NAVAL
POSTGRADUATE
SCHOOL

MONTEREY, CALIFORNIA

THESIS

MODELING AND ANALYZING INTRUSION ATTEMPTS TO A COMPUTER NETWORK OPERATING IN A DEFENSE-IN-DEPTH POSTURE

by

Mark Allen Givens

September 2004

Thesis Advisor: Alex Bordetsky
Co-Advisor: Joe Roth

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE				*Form Approved OMB No. 0704-0188*
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.				
1. AGENCY USE ONLY	**2. REPORT DATE** September 2004	**3. REPORT TYPE AND DATES COVERED** Master's Thesis		
4. TITLE AND SUBTITLE: Modeling and Analyzing Intrusion Attempts to a Computer Network Operating in a Defense in Depth Posture				**5. FUNDING NUMBERS**
6. AUTHOR: Mark A. Givens				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000				**8. PERFORMING ORGANIZATION REPORT NUMBER**
9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A				**10. SPONSORING/MONITORING AGENCY REPORT NUMBER**
11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.				
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited.				**12b. DISTRIBUTION CODE**

13. ABSTRACT *(maximum 200 words)*

In order to ensure the confidentially, integrity, and availability of networked resources operating on the Global Information Grid, the Department of Defense has incorporated a "Defense-in-Depth" posture. This posture includes the use of network security mechanisms and does not rely on a single defense for protection. Firewalls, Intrusion Detection Systems (IDS's), Anti-Virus (AV) software, and routers are such tools used. In recent years, computer security discussion groups have included IDS's as one of their most relevant issues. These systems help identify intruders that exploit vulnerabilities associated with operating systems, application software, and computing hardware. When IDS's are utilized on a host computer or network, there are two primary approaches to detecting and / or preventing attacks. Traditional IDS's, like most AV software, rely on known "signatures" to detect attacks. This thesis will focus on the secondary approach: Anomaly or "behavioral based" IDS's look for abnormal patterns of activity on a network to identify suspicious behavior.

14. SUBJECT TERMS Defense-in-Depth, Global Information Grid, Intrusion Detection, Intrusion Detection Systems, Layered Security, Network Security, Systems Security			**15. NUMBER OF PAGES** 109
			16. PRICE CODE
17. SECURITY CLASSIFICATION OF REPORT Unclassified	**18. SECURITY CLASSIFICATION OF THIS PAGE** Unclassified	**19. SECURITY CLASSIFICATION OF ABSTRACT** Unclassified	**20. LIMITATION OF ABSTRACT** UL

NSN 7540-01-280-5500 Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution is unlimited

MODELING AND ANALYZING INTRUSION ATTEMPTS TO A COMPUTER NETWORK OPERATING IN A DEFENSE-IN-DEPTH POSTURE

Mark A. Givens
Major, United States Marine Corps
B.S., Florida State University, 1991
M.S., Troy State University, 2000

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN INFORMATION TECHNOLOGY MANAGEMENT

from the

NAVAL POSTGRADUATE SCHOOL
September 2004

Author: Mark A. Givens

Approved by: Alex Bordetsky
Thesis Advisor

Joe Roth
Thesis Co-Advisor

Dan C. Boger
Chairman, Department of Information Sciences

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

In order to ensure the confidentially, integrity, and availability of networked resources operating on the Global Information Grid, the Department of Defense has incorporated a "Defense-in-Depth" posture. This posture includes the use of network security mechanisms and does not rely on a single defense for protection. Firewalls, Intrusion Detection Systems (IDS's), Anti-Virus (AV) software, and routers are such tools used.

In recent years, computer security discussion groups have included IDS's as one of their most relevant issues. These systems help identify intruders that exploit vulnerabilities associated with operating systems, application software, and computing hardware.

When IDS's are utilized on a host computer or network, there are two primary approaches to detecting and / or preventing attacks. Traditional IDS's, like most AV software, rely on known "signatures" to detect attacks. This thesis will focus on the secondary approach: Anomaly or "behavioral based" IDS's look for abnormal patterns of activity on a network to identify suspicious behavior.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

LIST OF FIGURES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGMENTS

The author would like to thank Professor Alex Bordetsky, LCDR Joe Roth, Mr. Robert Garza, and the Staff of the Network Security Group for making this research possible. I would also like to thank my wife Kelly, and children, Olivia and Ernie, for their support along this journey.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

> Defense in Depth: The DoD approach for establishing an adequate IA posture in a shared-risk environment that allows for a shared mitigation through: the integration of people, technology, and operations; the layering of IA solutions within and among IT assets; and, the selection of IA solutions based on their relative level of robustness.
>
> DoD Directive 8500.1, October 24, 2002

A. PROLOGUE

The intent of this chapter is to inform the reader of an aspect of information assurance that until recently, has not been considered a substantial problem. In addition, it will provide a description of a Defense-in-Depth posture and introduce the hypothesis. It will continue with discussion about Intrusion Detection Systems (IDS's), their placement, approaches, etc, and will discuss attack methodologies and attack types. It will conclude with a synopsis of the remaining chapters of this thesis.

B. BACKGROUND

In Greek mythology, it is believed that the Trojan Horse was an instrument of war used by the Greeks to access the city of Troy. In today's electronic society, the Trojan Horse is used by nefarious individuals to gain access to our most prized resource: information. As this "electronic war" proliferates, information security efforts will increase as System Administrators (SysAdmins) make haste to patch, harden, and secure their networks / systems. These laborious tasks are conducted in order to ensure the Confidentiality, Integrity, and Availability (CIA) of information that is processed, stored, or transmitted via Department of Defense (DoD) networks that operate on the Global Information Grid (GIG).

Although there is no "iron-clad" solution that keeps both malicious individuals off DoD networks or its information safe, there are methods employed that help curtail these intrusions. The DoD has formulated a modular, "Defense-in-Depth" strategy that builds upon multiple layers of network protection. This strategy is synonymous with Joseph Dell's "layered security model." Mr. Dell, Technology Director for InfoSecurity and Risk Management Consulting maintains:

> This model means you've got firewalls, IDS's, authentication servers, and encryption servers" [INFOSEC-03]. The DoD not only uses this approach, but also utilizes complementary technology as well. Specifically, DoD SysAdmins utilize "firewalls, intrusion detection and prevention systems, virus scanners, and content filtering devices that protect against not only external attacks but also internal threats as well" [NETSCREEN-03].

The Information Assurance Technical Framework (IATF) released the following four guiding principles for a Defense-in-Depth posture. It is these principles the DoD uses as guidelines for the Defense-in-Depth initiative.

1. Deploy Security Solutions Everywhere

Defense-in-Depth involves the deployment of protection mechanisms at multiple locations to resist all classes of attacks. When the network infrastructure is distributed, it is important to have proper security mechanisms at different areas to protect all networks from attacks [IATF-01].

2. Use Multiple Layers of Security Solutions to Protect the Network Against Intrusions and Attacks

Defense-in-Depth includes deploying multiple layers of defense between the adversary and his target. Multiple defenses include firewalls, intrusion detection and prevention, virus scanning, and other technologies, all working in parallel. Table 1 displays technologies available to implement each layer of defense [IATF-01].

3. Protect the Support Infrastructure

Networks, systems, and security mechanisms depend on a support infrastructure, which must be protected from adversaries. The support infrastructure includes elements such as Public Key Infrastructure (PKI), directory services, and user authentication infrastructure [IATF-01].

4. Collect and Analyze Security Events to Determine Threat Levels

Defense-in-Depth includes the continuous collection and analysis of intrusions and other security events. This information is used to determine the threat levels of network infrastructure, so that network administrators can properly and promptly react to changes in the threat levels and adjust the security posture of the network, if required [IATF-01].

Technology	Description
Firewall	Enforces access control on network traffic, selectively allowing external entities to access information protected by it. Firewalls are also used for denial-of-service (DoS) protection to defend networks against external or internal DoS attacks.
VPN	Provides confidentiality and integrity to the data transmitted across a public network. VPNs also facilitate the implementation of communities of interest (COIs)
Intrusion Detection and Prevention (IDP)	Detects and blocks network attacks. IDP systems use knowledge of higher level protocols and applications to identify network attacks
Content Filtering	Performs content checking mechanisms for passing data, including anti-virus detection and protection
Public Key Infrastructure (PKI)	Authenticates users, devices and applications when sending, receiving or accessing information.

Table 1. Defense-in-Depth Technologies

C. HYPOTHESIS

The inclusion of an anomaly based Intrusion Detection System has no significant effect on the security posture of a network that operates in a Defense-in-Depth environment. This statement arises because traditional security experts believe that anomaly based IDS's provide too many "false positives" to be considered helpful. They maintain that signature based IDS's, when working in unison with other layered security devices, handle the majority of the workload, therefore rendering the anomaly based IDS useless.

1. Ho

An anomaly based IDS has no prolific impact on a network's security posture because the other complementary security devices provide adequate redundancy.

2. Ha

There is a measurable "value added" of anomaly based IDS's compared to networks that solely use signature based IDS's.

This thesis will prove the Null Hypothesis statistically insignificant. It will use $\alpha = .05$ (significance level) to provide evidence in favor of the alternative hypothesis.

The premise of this thesis is to analyze the theory quantitatively that networks operating on the GIG must adhere to a Defense-in-Depth posture. Specifically, its purpose is to focus on the aforementioned second principle that pertains to the implementation of IDS's as layered devices in a Defense-in-Depth posture. The intent of this thesis' experiment is to collect data from a network that incorporates both signature and anomaly based IDS's as the networks layered security devices. It will justify the "value added" of an anomaly based IDS that operates in conjunction with a signature based IDS.

Figure 1 is an organizational model of a single layer line of defense and Figure 2 is an organizational model of a multilayered network that operates in a Defense-in-Depth posture.

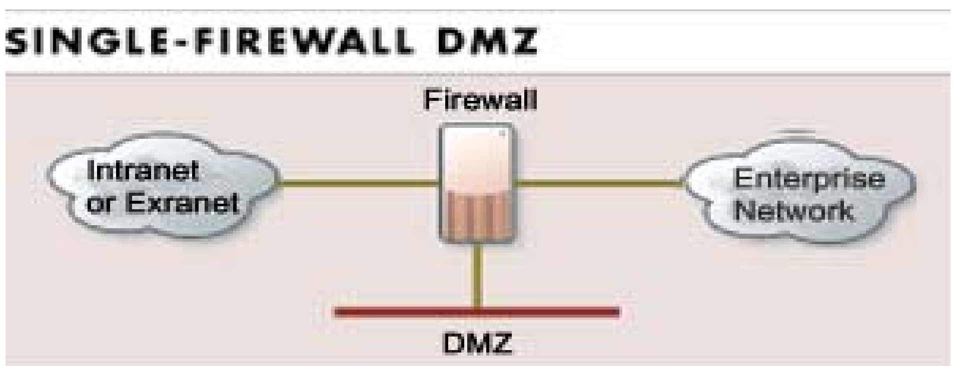


Figure 1. Single Layer of Defense Model

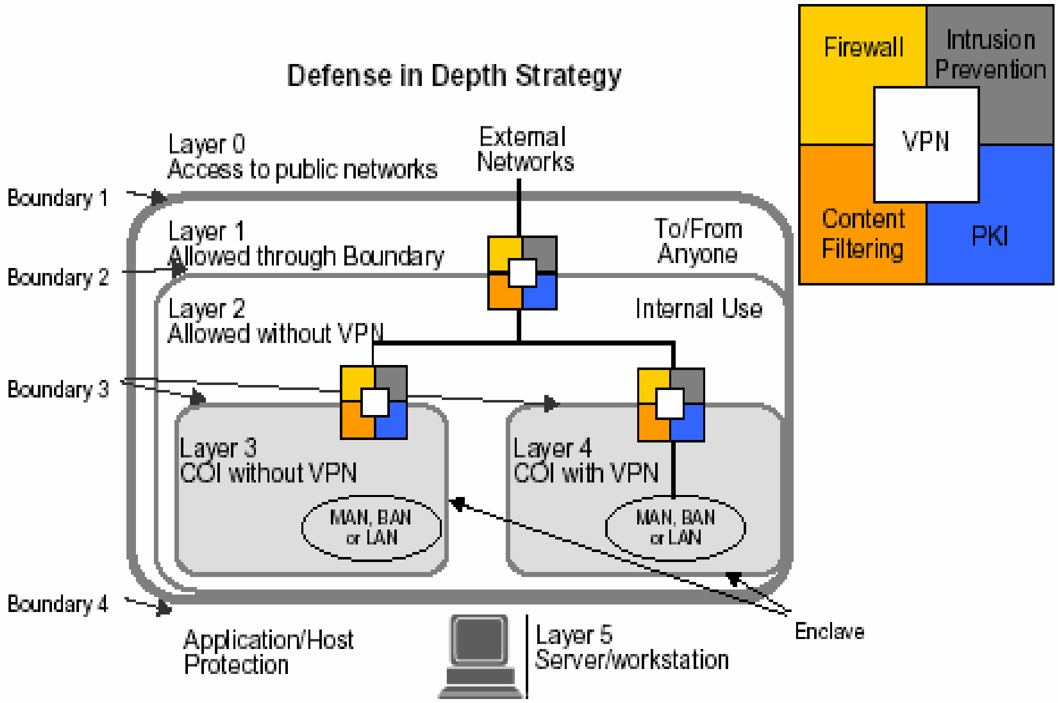


Figure 2. Defense-in-Depth Model Used By DoD [From: NetScreen]

D. IDS DESCRIPTION

In order to describe an IDS, intrusion detection must first be defined: It is the process of monitoring events occurring in a computer system or network and analyzing them for signs of *intrusions,* defined as attempts to compromise the confidentiality, integrity, availability, or to bypass the security mechanisms of a computer or network. Intrusions are caused by attackers accessing the systems from the Internet, authorized users of the systems who attempt to gain additional privileges for which they are not authorized, and authorized users who misuse the privileges given them. Therefore, an IDS is defined as: "software or hardware products that automate this monitoring and analysis process" [BACE-00].

E. IDS TYPES

Intrusion detection systems comprise both hardware appliances and software applications. The majority of IDS's are network based systems. They are either stand alone appliances or network sniffing software. These types capture and analyze information packets as they traverse a network. Their complement, host-based systems, reside as application software on a host computer. Furthermore, application based IDS's are a subset of Host-based IDS's. These types analyze events that occur within the application software. "Programs can scan computer records or on-line computer activity for patterns that indicate or suggest the presence of unauthorized activity" [DENN-02].

F. IDS APPROACHES

There are two approaches used when detecting intrusion attempts.

1. Signature Based IDS

Recognizes a known signature or pattern that resides in a local knowledge base established by the vendor. Periodically, IDS vendors issue signature "update" messages so the consumer can update his local knowledge base. These systems can be configured to automatically check the vendor website for updates and install them accordingly. Bace refers to these types of systems as "misuse" detectors.

2. Anomaly or "Behavioral" Based System

> Anomaly detectors identify abnormal unusual behavior (anomalies) on a host computer or network. They function on the assumption that attacks are different from "normal" activity and can therefore be detected by systems that identify these differences. Anomaly detectors construct a normal behavior profile of authorized / legitimate users, hosts, or network connections. These profiles are constructed from historical data collected over a period of normal operation. The detectors then collect event data and use a variety of measures to determine when monitored activity deviates from the norm [BACE-00].

G. IDS PLACEMENT

IDS's can operate as independent, stand alone, network based systems on either side of a network firewall device. Network based IDS's identify and prevent intruders that exist external to a network or recognize internal threats as well. IDS's also operate as host-based systems that reside on a local computer and are generally application software. These types of systems identify attempted privilege escalation or those user characteristics outside normal parameters (obscure web sites, unauthorized access, etc).

H. ATTACK TYPES

Table 2 describes the five classes of attacks that IDS's should be able to detect:

Attack	Description
Passive	Passive attacks include traffic analysis, monitoring of unprotected communications, decrypting weakly encrypted traffic, and capture of authentication information (e.g., passwords). Passive intercept of network operations can give adversaries indications and warnings of impending actions. Passive attacks can result in disclosure of information or data files to an attacker without the consent or knowledge of the user. Examples include the disclosure of personal information such as credit card numbers and medical files.
Active	Active attacks include attempts to circumvent or break protection features, introduce malicious code, or steal or modify information. These attacks may be mounted against a network backbone, exploit information in transit, electronically penetrate and enclave, or attack an authorized remote user during an attempt to connect to an enclave. Active attacks can result in the disclosure or dissemination of data files, denial of service, or modification of data.
Close-In	Close-in attacks consist of a regular type individual attaining close physical proximity to networks, systems, or facilities for the purpose of modifying, gathering, or denying access to information. Close physical proximity is achieved through surreptitious entry, open access, or both.
Insider	Insider attacks can be malicious or nonmalicious. Malicious insiders intentionally eavesdrop, steal or damage information, use information in a fraudulent manner, or deny access to other authorized users. Nonmalicious attacks typically result from carelessness, lack of knowledge, or intentional circumvention of security from such reasons as "getting the job done."
Distribution	Distribution attacks focus on the malicious modification of hardware or software at the factory or during distribution. These attacks can introduce malicious code into a product, such as a backdoor to gain unauthorized access to information or a system function at a later date.

Table 2. Attack Types

In addition to these preceding descriptions, several additional attack types warrant attention. In his thesis "Using Network Management Systems to Detect Distributed Denial of Service Attacks," Chandan Negi discusses the following attacks and methodologies.

1. Denial of Service Attack

Characterized by an attacker trying to prevent the use of resources by legitimate or authorized users. It is a "dominating conversation with a network resource designed to preclude other conversations with that resource" [STROT-00].

Examples include:

- attempts to "flood" a network, thereby preventing legitimate network
- traffic
- attempts to disrupt connections between two machines, thereby
- preventing access to a service
- attempts to prevent a particular individual from accessing a service
- attempts to disrupt service to a specific system or person [NEGI-01]

2. Distributed Denial of Service Attack

Amplified by adding a "many to one" relationship to these attacks, making them more difficult to prevent [NEGI-01].

I. ATTACK METHODOLOGIES

1. Denial of Service Attack Methodologies

Table 3 displays widely used DOS methods. These types of attacks take advantage of flaws associated with network protocols.

Attack	Description
Bonk	This attack exploits a "lack of bounds" defect associated when reassembling IP packets. This attack occurs because fragments were sent with offsets that do not align.
Ping of Death (PoD)	Directed towards the Internet Control Message Protocol (ICMP) echo request. ICMP is used with a small payload to provide a fast, low bandwidth test of connectivity. Because of this typical usage, some software applications do not handle large payloads. If such software receives an ICMP request packet with a payload greater than 4000 bytes the software generates an exception and halts communication on the network.
Smurf Attack/Syn Flood	A network security breach in which a network connected to the Internet is swamped with replies to ICMP echo (PING) requests. A Smurf attacker sends PING requests to an Internet broadcast address. These are special addresses that broadcast all received messages to the hosts connected to the subnet. Each broadcast address can support up to 255 hosts, so a single PING request can be multiplied 255 times. The return address of the request itself is spoofed to be the address of the attacker's victim. All the hosts receiving the PING request reply to this victim's address instead of the real sender's address. A single attacker sending hundreds or thousands of these PING messages per second can fill the victim's T-1 (or even T-3) line with ping replies, bringing the entire Internet service to its knees.
UDP Flood	This attack is based on UDP echo and character generator (chargen) services. Forged UDP packets are used to connect the 'echo service' on one machine to the 'chargen service' on the other machine. The result is that the two services consume all available network bandwidth between the machines as they exchange characters between themselves.
Land Attack	The source and destination addresses are identical, therefore the "victim" consumes all its resources talking to itself, thereby obstructing outside conversations.

Table 3. Denial of Service Attack Methodologies [From: NEGI-01].

2. Distributed Denial of Service Attacks Methodologies

Table 4 describes the methods employed by attackers that utilize Distributed DOS techniques.

Attack	Methodology
Trinoo	The conversation between the master and the slave(s) uses TCP, while the conversation with the attack daemons uses UDP. The implemented attack is a UDP flood.
Tribe Flood Network (TFN)	Conversations between the master and the slave(s) uses ICMP echo reply packets. The type of attack could be variations of Smurf, SYN flood, UDP flood, and ICMP flood attacks.
TFN2K	This is the newer version of the TFN attack and uses TCP, UDP, ICMP, or all three to communicate between the master and the slave(s) and the communication is encrypted. The attacks implemented are the same as TFN.
Stacheldracht	This is based on the TFN attack. The conversation between master and slave(s) is encrypted and uses TCP and ICMP. Implemented attacks are the same as TFN.
Shaft	Modeled after Trinoo. Conversation between master and slave(s) is achieved using UDP packets and implemented attack is the UDP flood attack. An important feature of this attack is its ability to switch control master servers and ports in real time, thereby making the detection by intrusion detection tools difficult.

Table 4. Distributed Denial of Service Attack Methodologies [From: NEGI-01].

Although this is not a totally inclusive list of attack types or methodologies, mainly because there are many not yet detected or developed, its intention is to enlighten the reader to some currently known.

J. INTRUDERS

Intruders can fall in two broad categories: Those individuals considered “outside” of an organizations network and those currently employed are considered “inside.” Most people perceive the outside intruder as the biggest threat. The media scare about “hackers,” “crackers,” and “attackers” attempting access over the internet has heightened this perception. Recent FBI studies revealed that ninety percent of those U.S. companies that experienced internet fraud in a two-year period were victims of the worst intruder type, the Insider [YUN-01].

K. ORGANIZATION

The remainder of this thesis is organized as follows. Chapter II will discuss the roles of those complementary devices required in a Defense-in-Depth posture. It will furthermore provide a preliminary view of the planned intrusion experiment. Chapter III will describe the conduct of this thesis' experiment and gather the necessary evidence needed to substantiate the alternative hypothesis. Chapter IV will analyze and discuss the data captured during the experiment. Chapter V will analyze and make statistical inferences on the collected data. It will prove either the null or alternative hypothesis as statistically significant. Chapter VI offers conclusions and recommendations for further work and the Appendix is an article from InfoWorld Magazine dated 26 August 2004. This involves an IDS comparison with which the author was involved.

L. CHAPTER SUMMARY

In summary, this chapter exposed the reader to problems that occur when networks are left unguarded. A Defense-in-Depth posture is needed to ensure the CIA of information.

THIS PAGE INTENTIONALLY LEFT BLANK

II. PROBLEM PROPOSAL, COMPLEMENTARY IDS DEVICES, AND EXPERIMENT OVERVIEW

> Whereas access controls and filters seek to prevent unauthorized or damaging activity, intrusion and misuse detection mechanisms aim to detect it at its outset or after the fact.
>
> Dorothy Denning, *Information Warfare and Security,* 2002

A. PROLOGUE

This chapter will present the reader with a significant problem encountered when only a signature based Intrusion Detection System (IDS) is deployed with other layered security devices. Additionally, it will offer general descriptions of those complementary devices that work with both Host and Network Based IDS's. It will conclude with an overview of this thesis' planned intrusion detection system experiment.

B. SYNOPSIS OF IDS PROBLEM

Information assurance professionals, both DoD affiliated and those that perform their duties in the commercial arena, must comprehend the "value added" of incorporating anomaly based IDS's in conjunction with other layered security devices, to include signature based IDS's.

Network defenses that are established with signature based IDS's will only identify deterministic, predefined patterns of attack. For example, the SNORT IDS currently has over 2300 identified signatures in its knowledge base. This is all well and good for those known attacks, but what about those attacks that are newly written, not yet discovered, or polymorphic? With a signature based IDS, the known signature is compared to the datagram as it passes by the IDS. If the knowledge base is current with signatures, known attacks are identified at this point. If the knowledge base is not up to date, then the intrusion is allowed to pass.

However, if an anomaly based IDS is employed with other complementary boundary defenses, any conversation that is out of a statistically normal pattern will be flagged as an intrusion and an appropriate warning will be sent. These warning messages are in the form of audible noises, e-mail alerts, and pop up messages.

C. DEFENSE-IN-DEPTH DEFINITION

The Department of Defense (DoD) uses the term "Defense-in-Depth" to describe its approach to Information Assurance (IA). This approach integrates the capabilities of people, operations, and technology to establish a multi-layer and multi-dimensional protection of DoD assets. The Defense-in-Depth approach builds mutually supporting layers of defense to reduce vulnerabilities and help protect against, detect, and react to as many attacks as possible. The goal of this defense is to cause an intruder who penetrates or breaks down one defensive layer to encounter multiple barriers promptly until the quest for unauthorized entrance ends [DISA-01].

In order to ensure the success of a Defense-in-Depth posture, the DoD recommends defense of the following four elements: Local Computing Environments (also known as Enclaves); Enclave Boundaries; Networks that link Enclaves; and Supporting Infrastructures.

1. Local Computing Environment - (Enclave)

The Enclave is comprised of those computing assets that are utilized within an organization. This includes data and information, software applications (commercial and proprietary), data processing technology, personnel, and those facilities that reside under a singular authoritative, security-related uniform corporate policy.

Items that comprise an enclave include: Intranet, service layer networks, Secret Internet Protocol (IP) Router Network (SIPRNet), Remote LANS or systems, and Virtual Private Networks (VPN).

2. Enclave Boundaries

These points connect the local area network (LAN) to the Internet. This is where the de-militarized zone (DMZ), firewalls, routers, network based IDS's, and proxy servers are established. The defense of the enclave boundary should include protecting data integrity as it traverses the network, protecting the physical and logical boundaries of the enclave, and protecting the availability of those systems and networks that operate internal to an enclave.

3. Enclave Linked Networks

At the network level, the transport mechanisms used for user traffic is the focus of Defense-in-Depth. These mechanisms include transmission and switching capabilities.

Examples of networks include the Non-Classified IP Router Network (NIPRNet), SIPRNet, the Joint Worldwide Intelligence Communications System (JWICS), and the Defense Information Systems Network (DISN).

4. Supporting Infrastructures

These provide security services for networks, enclaves, and computing environments. Examples of supporting infrastructures include: cryptography and key management; incident detection, reporting, and response.

D. COMPLEMENTARY ENCLAVE BOUNDARY DEFENSES

Enclave Boundary Defenses protect those services and data from lurking outlying dangers. They also protect those elements within an enclave that do not protect themselves [DISA-01]. The technologies associated with defending enclave boundaries include antivirus protection, content filtering devices, firewalls, identification and authentication devices, intrusion detection, intrusion prevention, malicious code detectors, proxy servers, virtual private networks, public key infrastructure, SPAM blocking devices, Secure Shell, and IPSec. Enclave boundary defenses are the focus of this thesis.

1. Anti-Virus Protection (AV)

AV protection programs are utilities that search boot sector blocks, hard disks, mail programs, executable files, or application software for viruses and either deletes or quarantines those found. Most AV programs include auto-update and auto-scan features. Automatic update features enable the software kernel to download profiles or "signatures" of new viruses while automatic scan enables the ability to check for those newly discovered viruses. AV software is designed to establish a baseline or signature knowledge base for the system files and application software and regularly monitor and verify that their integrity is maintained.

2. Content Filtering Devices

These devices, also known as content security devices, apply security policies to the content or "payload" of a datagram. This collectively refers to AV, content monitoring (URL filtering), and e-mail filtering. Unlike infrastructure elements, such as routers, firewalls and many IDS's that look at content independent of context, a content filtering device must reassemble the datagram before the content can be analyzed.

3. Firewalls: (Gateways)

Firewalls consist of network appliances (personal computers, routers, or dedicated hardware) or host-based application software that filters all inbound traffic from untrusted sources into a LAN or other private system. These systems can also be used as access control mechanisms to filter those personnel within an organization seeking access to the Internet. These devices rely on information that is generated at all levels of the Open Systems Interconnection (OSI) model. "As a rule, the higher the OSI layer at which your firewall examines these packets, the greater the protection provided" [MAIRS-02].

Hardware firewall products protect your computer and home network by guarding your Internet connection and filtering any requests that you have not specifically allowed.

Software firewalls are installed directly on your PC, and filter requests after they reach your computer. These are often less expensive and easier to configure than hardware firewalls. They furthermore ease the burden of having to reconfigure the abundant "spaghetti" coil of cables whenever a new system is incorporated. Software firewalls provide more assurance than simple router firewalls because they provide additional protection from spyware and Trojan horses. If you travel with a laptop, a software firewall is a necessity—you need protection wherever you connect to the Internet, and your hardware firewall can protect you only at home [NORTH-02].

There are basically four types of firewalls.

a. Packet Filtering Gateways

These are the most basic form of firewalls. They work by applying rule set filters against packets of data that traverse networks. A packet filtering firewall regulates traffic flow based on TCP/IP header information. The packets that pass through the filters are sent to the requesting system and all others are discarded. There are two types of packet filtering firewalls: Stateless and Stateful.

- Stateless firewalls evaluate each datagram and maintain no "state" of conversation.
- Stateful filters "listen to all communications and store these conversations in memory" [MAIRS-02]. Packet filtering firewalls generally only check information at OSI layer 3.

b. Circuit Level Gateways (CLG)

These firewalls act as the conduit that connects an external enclave source computer to an internal LAN destination host. This is accomplished by the source first establishing a connection to an available CLG port and then the CLG connects to the intended destination computer. Acting as middle-man, the CLG copies verbatim all data bytes from the source to the respective destination. Internal network information is concealed from outside sources because it appears the CLG is the originator. These firewalls filter packets up to OSI Layer 5.

c. Application Level Gateways

These firewalls examine all traversing packets. However, they first copy and then forward all packets through the firewall by performing as a proxy. Since this is an OSI Layer 7 event, only those specific protocols can copy, filter, or forward individual type traffic. For example, if a file transfer protocol (FTP) application level firewall is being used, only FTP traffic is allowed to pass through the firewall, all others are rejected.

d. Stateful Multi-Layer Inspection (SMLI) Gateways

These firewalls are combinations of the previously mentioned gateways in paragraphs a, b, and c. They allow legitimate users direct access to the internet by maintaining dynamic state tables on every conversation made. They inspect packets at Layer 3 and also inspect the contents of Layer 7 as well. They are considered to be "Stateful" firewalls, meaning they remember characteristics of data packets that traverse the network. Although they require more micro-processing time, these firewalls compare received packets with those saved, and then decide on datagram passage. In his book, "*VPN's, A Beginner's Guide*," Mairs refers to this as a "direct transparent connection" between a client and host.

4. Identification and Authentication Devices

Identification and authentication tools are used as recognition devices for those remote users requesting enclave access. These control mechanisms perform their duties by verifying Personal Identification Numbers (PIN's), strong passwords, the various forms of biometrics, and electronic tokens.

5. Intrusion Detection Systems

Described in Chapter I.

6. Intrusion Prevention Systems (IPS's)

IPS's are considered the next logical step in the evolution of IDS's. These systems are the combination of the blocking capabilities of firewalls with the deep packet inspection capability of IDS's. An IPS is defined as "any device (hardware or software) that has the ability to detect attacks, both known and unknown, and prevent the attack from being successful" [DESAI-03]. IPS's are emerging technologies, and as of this writing research is ongoing.

7. Malicious Code Detectors

Malicious code is usually classified according to both its propagation method and goal [McGRAW-00]. Table 5 details malicious code types:

Type	Description
Viruses	Programs that self-replicate within a host by attaching themselves to programs and/or documents that become carriers of the malicious code.
Worms	Self-replicate across a network.
Trojan Horses	Masquerade as useful programs but contain malicious code to attack the system or leak data.
Back Doors	Open the system to external entities by subverting the local security policies to allow remote access and control over a network.
Spy-Ware	Is a useful software package that also transmits private user data to an external entity.

Table 5. Malicious Code Types

Malicious code detectors are those software applications that possess the ability to process, screen, and identify malevolent datagram packets that bi-directionally traverse a network. These systems, usually Stateful in nature, must be placed at the Enclaves boundary in order to identify and destroy harmful code.

8. Proxy Servers

Proxy servers perform two functions.

a. IP Address Translator

A firewall that uses a process called "address translation" to map all of your internal IP addressees to one IP address that is externally visible to those outside your network. This address is associated with the firewall from which all outgoing packets originate [MAIRS-02].

b. Request Filter

Proxy servers can also be used to filter requests. For example, a company might use a proxy server to prevent its employees from accessing a specific set of Web sites. This capability is particularly useful when such sites are known sources of malicious code or other hostile action [WEB-03].

9. Public Key Infrastructure (PKI)

PKI involves the use of the asymmetric private – public key pair and ensures that confidentiality, authenticity, and integrity of a message is maintained. PKI algorithms allow for digital signatures which provide three important security services:

a. Integrity

Any change to the contents of a message during transport results in message integrity failure.

b. Non-Repudiation

Only the holder of a private key can digitally sign a message. Since each private key is unique to an individual, this cannot be refuted.

c. Authentication

This is a way to identity if the sender of an electronically transmitted message is legitimate. The recipient can be assured that the messages origin and author are bona fide.

10. Virtual Private Networks (VPN)

A VPN is a "virtual" network that is kept private by "tunneling" private data through the underlying infrastructure of the public Internet [MAIRS-02]. VPN's achieve security through "end-to-end" authentication and encryption. Data packets are encapsulated within network protocols that are understood at both the sending and receiving ends of transmission. This is referred to as the VPN tunnel. The entry point system encapsulates the data packets while the exit point removes the data from the encapsulated datagram. VPN's create virtual circuits by special tunneling protocols and must encapsulate each source network packet into a packet that contains the connection management intelligence necessary to establish and disassemble the tunnel.

VPN tunnels are based on two types of protocols: The first, established at Layer 2, uses frames as their exchange unit. Layer 2 tunneling provides extra protection by encrypting all of each datagram except the link-level information. This prevents a listener from obtaining information about network structure. While this encryption prevents traffic analysis, the datagram must be encrypted and/or decrypted on every network hop.

The second tunneling protocol is established at Layer 3 and uses packets as the transfer mule. Layer 3 protocols encapsulate IP packets into an additional header before transfer occurs.

11. SPAM Blocking Devices

SPAM is defined as the posting of irrelevant or inappropriate messages to one or more Usenet newsgroups, mailing lists, or other messaging system in deliberate or accidental violation of netiquette. It is basically flooding the Internet with many copies of the same message. Spammers are attempting to force their message on people who would not otherwise choose to receive it. Fortunately, for the consumer, there are various SPAM control devices (hardware and software) that are available on the commercial market. Favored choices are Bayesian filtering and heuristics, followed by signature/content matching, and blacklisting [INFOSEC-04].

12. Secure Shell (SSH)

A Unix shell program for logging into and executing commands on remote computers. SSH is intended to replace "rlogin" and "rsh," and provide secure encrypted communications between two untrusted hosts over an insecure network. X11 connections and arbitrary TCP/IP ports can also be forwarded over the secure channel [WEB-03].

13. IPSEC

A protocol that provides security for transmission of sensitive information over unprotected networks such as the Internet. IPsec acts at the network layer, protecting and authenticating IP packets between participating devices ("peers") [WEB-03].

E. EXPERIMENT OVERVIEW

The test bed that will be used for the IDS comparison experiment consists of several desktop computers and IDS servers that operate on a single Fast Ethernet network. This LAN is configured with broadband Internet access via a satellite transmitting/receiving antenna. These computer hosts will be configured with various operating systems and application software in such a manner as to monitor and collect information from those nefarious attempts that probe or penetrate the networks defenses.

The networks architectural defense strategy will model a Defense-in-Depth posture. In order to measure the effectiveness of this strategy, quantifiable instances of intrusions or penetrations must be incurred. The experiment's goal is to analyze those malicious activities identified by both the signature based and anomaly based IDS's. The hosts will be configured with complementary defensive devices. However, the outer perimeter of the network will host both the signature and anomaly based IDS servers. These servers will monitor all traffic that bi-directionally traverses the network. The experiment's analysis phase's intent is to interpolate the differences associated with both types of IDS's.

1. Milestones

The experiment's start date is 1 May 2004 and will run consecutively for a time period long enough to collect quantifiable data, tentatively 19 June. This will allow a 50 day observation / collection period. Various methods that detect and capture probes and other malicious activity will be used. Upon completion, an analysis of the collected data will begin.

F. CHAPTER SUMMARY

This chapter introduced the reader to a significant problem encountered when only a signature based IDS is used. Furthermore, it presented general descriptions of those complementary devices that work with IDS's in a Defense-in-Depth posture. It was not this chapter's intent to delve deeply into their individual abilities or characteristics. It concluded with a brief experiment overview.

THIS PAGE INTENTIONALLY LEFT BLANK

III. TEST BED SETUP AND CONFIGURATION

> Any intelligent fool can make things bigger, more complex, and more violent. It takes a touch of genius -- and a lot of courage -- to move in the opposite direction.
>
> Albert Einstein

A. PROLOGUE

The intent of this chapter is to detail the IDS experiment configuration. It will list the arrangements of the hardware, software, and network components that comprise the experimental lab.

The network is comprised of desktop computers and network IDS servers that will maintain an operational working status for the entire projected test time.

B. HARDWARE AND SOFTWARE CONFIGURATION

The equipment that will be utilized in this experiment consists of six Intel based personal computers and one Dell 1U rack-mountable server that all operate on one Fast Ethernet network. The network is furthermore described.

1. Internet Access

Internet access is granted via the DIRECWAY[r] broadband Internet Service Provider (ISP). This connectivity is a bidirectional satellite antenna that operates at 1370 megahertz with upload/download rates of 899/60, respectively. See Table 6 for details. Since this is a broadband connection, its operational status is expected to be continuous for the entire experiment period. The MODEM that is used with this system has the static IP address of 67.44.111.177. This is a FIRMWARE IP address and cannot be manipulated.

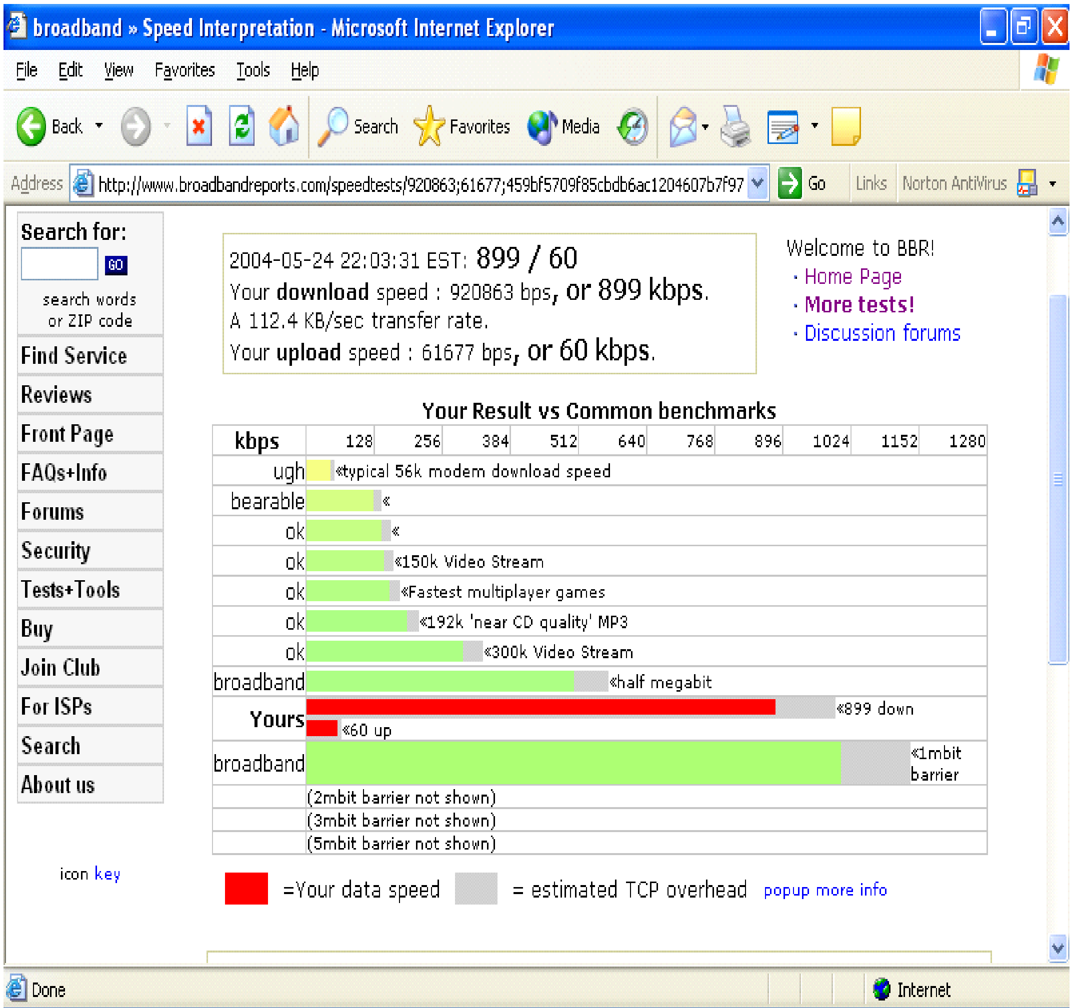


Table 6. Broadband Reports Speed Test

2. Router, Switch, Hubs, and Cables

The gateway for the network is the NETGEAR® Cable / DSL Web Safe Router # RP614. This is a four port router with 10/100 Mbps automatic sensing switch. It comes preconfigured with TCP/IP networking to include network address translator (NAT) and port forwarding. The router has been enabled to keep access logs and will periodically send those logs to a Simple Mail Transfer Protocol (SMTP) account. The wide area network (WAN) IP address is 67.44.111.178 while the local area network (LAN) address is 10.11.12.1.

The network switch that is employed is the NETGEAR® FS605 Fast Ethernet 5 port switch. Two hubs will be used, the NETGEAR® DS108 10/100 dual speed 8 port hub and the ASOUND® 10/100 dual speed 8 port petit hub.

Additionally, category 5 (CAT 5) cable will be used. This will help ensure that a 100 Mbps flow rate is achieved.

The LAN Internet Protocol (IP) address range consists of the non-routable Class C address space of 10.11.12.0/24 and is designated as the STOOGE-CENTRAL network.

3. Desktop Computers and Servers

The following computers connect directly to the router. Their configurations are as follows:

a. Computer 1 [MAGIVENS]

- P4 - 2 GHZ microprocessor, 992 MB of RAM
- IP Address: 10.11.12.6
- Windows XP[r] Professional with Service Pack 1
- SYMANTEC[r] Norton AntiVirus 2004
- ISS BlackIce[r] Defender
- STOPZilla[r] pop up protection

b. Computer 2 [SWEETKELLY]

- P4 - 1.8 GHZ microprocessor, 128 MB of RAM
- IP Address: 10.11.12.2
- Windows XP[r] Professional with Service Pack 1
- SYMANTEC[r] Client Firewall
- NORTON AntiVirus Corporate Edition
- ISS BlackIce[r] Defender
- STOPZilla[r] pop up protection

The following computers connect to the switch. Their configurations are as follows:

c. Computer 3 [SAMBA SERVER]

- P4 – 2 GHZ microprocessor, 512 MB of RAM
- IP Address: 10.11.12.7
- Linux RedHat 9
- SAMBA Server

d. Computer 4 [LARRY]

- P2 – 450 MHZ microprocessor, 256 MB of RAM
- IP Address: 10.11.12.4
- Windows XP[r] Professional with Service Pack 1
- SYMANTEC[r] Client Firewall
- NORTON AntiVirus Corporate Edition

e. Computer 5 [CURLY-SERVER]

- P2 – 450 MHZ microprocessor, 256 MB of RAM
- IP Address: 10.11.12.3
- Windows[r] 2000 Server with Service Pack 3
- FTP Server
- IIS Server
- Microsoft SQL Server with Service Pack 3

4. Enclave Boundary Defense Configuration

It is the intent of this experiment to monitor and measure those nefarious attempts to penetrate the STOOGE-CENTRAL network. Therefore, two network based IDS's will be used. They will be incorporated as the network's perimeter's defense mechanisms and will capture all traffic that traverses the network. Their configurations are detailed below.

a. Pentium[r] Server [MOE]

- P2 – 450 MHZ microprocessor, 256 MB of RAM
- IP Address: 10.11.12.5 (management NIC)
- 2nd NIC connected to Hub for traffic monitoring
- Still Secure Border Guard[r] Enterprise Edition Signature Based IDS Software

b. Dell 1750 Server

- Lancope StealthWatch[r] Anomaly Based IDS
- IP Address: 10.11.12.10 (management NIC)
- 2nd NIC connected to Hub for traffic monitoring

Figure 3 below portrays this network and Figures 4 and 5 display a vulnerability assessment scan taken by Shields Up found at https://grc.com.

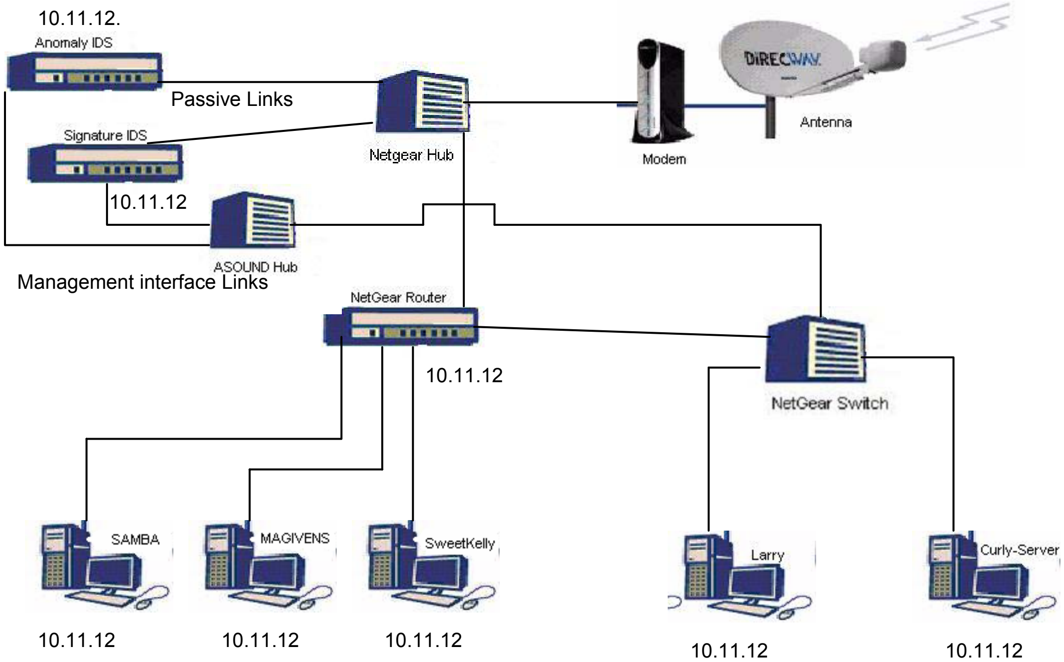


Figure 3. IDS Experiment Configuration

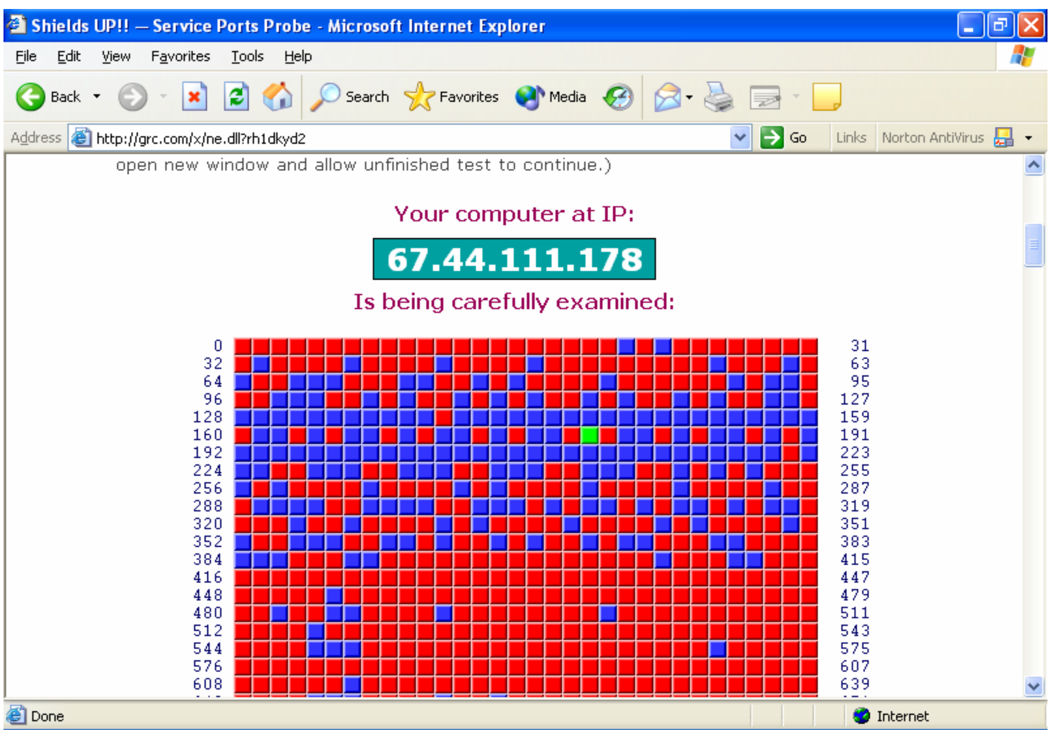


Figure 4. Shields Up Vulnerability Assessment Part A

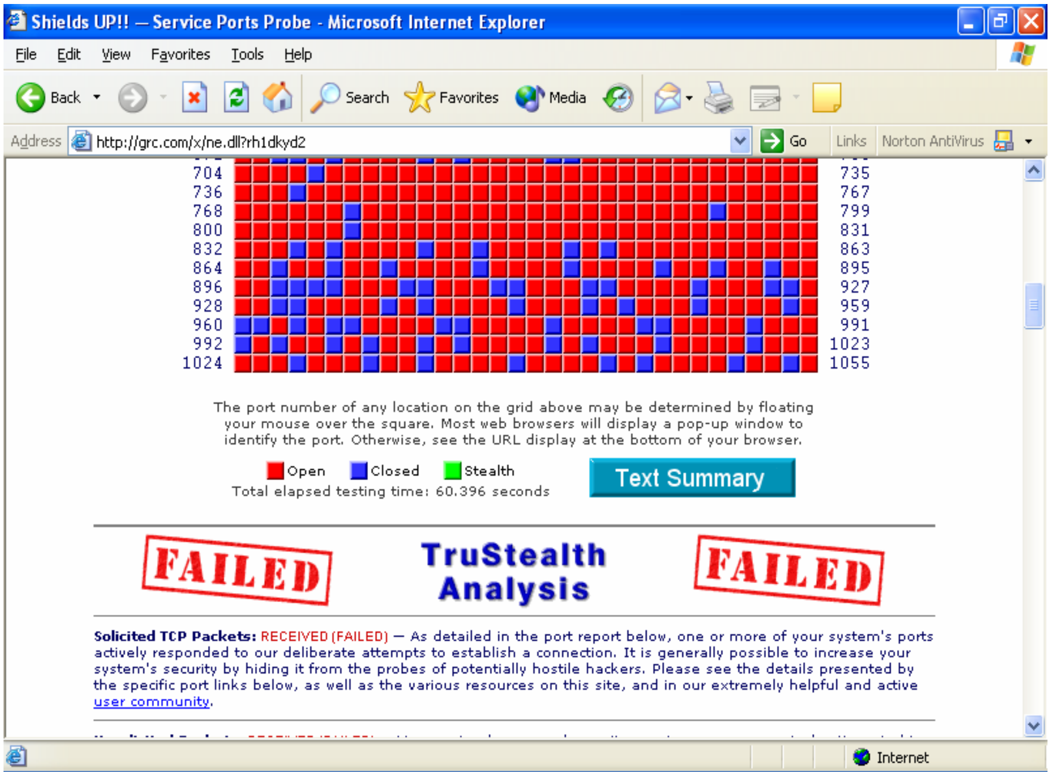


Figure 5. Shields Up Vulnerability Assessment Part B

C. STOOGE-CENTRAL IDS'S OVERVIEW

This section will detail the characteristics about the two IDS servers.

1. Signature Based IDS

The signature based IDS detects malicious traffic through a patented process called Dynamic Attack Detection (DAD) technology [BG-03]. This is a combination of rule-set methods that utilize the Open-Source signature based IDS software, SNORT. The signature based IDS's foundation is the SNORT based rule-sets, however, they have been greatly enhanced in both performance and CPU efficiency. It uses the following methods to detect intrusion attempts.

a. Stateful Packet Analysis

This is achieved through SNORTS Stream4 and Frag2 pre-processors. Stateful inspection and analysis prevents mischievous intrusion attempts and increases the accuracy of alerts, thus helping to alleviate "False Positives."

b. Signature Analysis

The signature based IDS maintains a knowledge-base of known signatures obtained from the Open-Source arena and those crafted by the user. These signatures provide the necessary pattern matching capability and content analysis needed to detect known attacks.

c. Protocol/Anomaly Analysis

The signature based IDS's protocol anomaly detectors build models of transmission control protocol (TCP) IP protocols using specifications / request for comments (RFCs). This detection method ensures that events within a session conform to the proper state as defined by the protocol. Protocol / anomaly analysis is used to detect a wide range of known and unknown attacks [BG-03].

d. Layer 2 Analysis

Detects invalid and malicious activity at OSI layer 2 to include both duplicated and spoofed media access control (MAC) addresses.

2. Signature Based IDS Functionality

The following describes the signature based IDS's basic functions and is taken from vendor literature:

a. Detect

An attack is launched against your network and the signature based IDS identifies and stops the attack.

b. Qualify and Respond

The signature based IDS's response is based on its configuration:

- block the attack
- alert and prompt
- block the attack by a responsive policy or custom command script
- ignore this instance
- take action at a later time.
- ignore the attack – If configured, it will ignore the attack. This is allowed if certain hardware or applications are not installed on your network

c. Manage and Report

Analyze and report on attack activity. It sends e-mail alerts and logs all attack activity and produces various reports.

This is further illustrated in Figure 6.

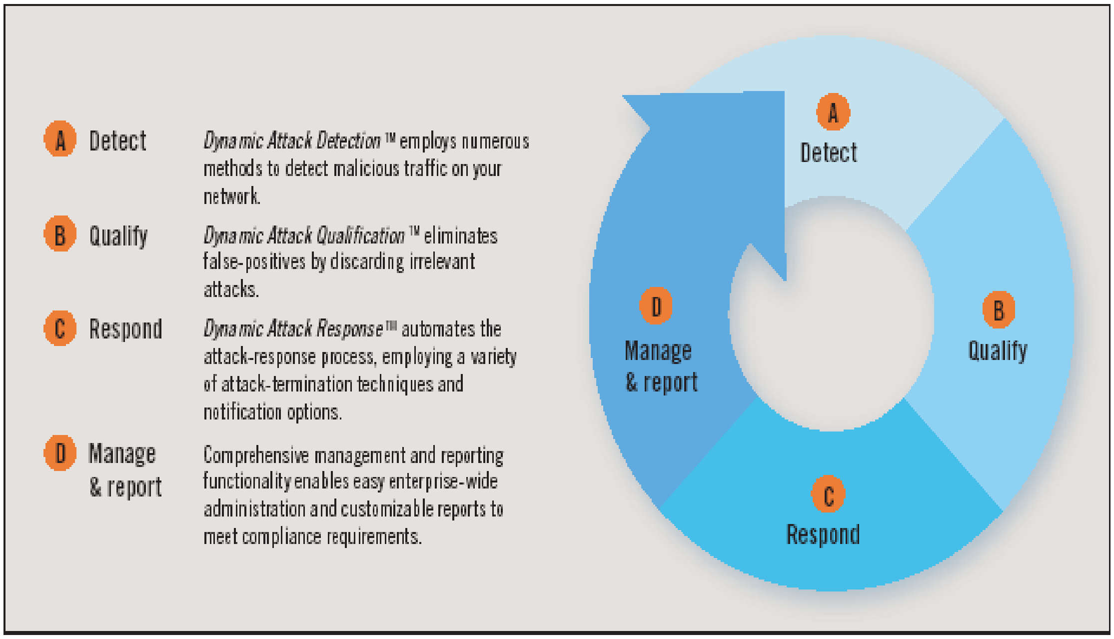


Figure 6. Signature Based IDS's Detection Cycle [From: BG-03]

3. Anomaly Based IDS

The anomaly based IDS monitors, detects, and responds to unwarranted access attempts and internal misuse on networks. The appliance recognizes zero-day attacks, responds with various alarm mechanisms, and creates forensic data of network activity.

The appliance approaches intrusion detection/prevention through a behavior-based architecture that responds to statistical anomalies that occur within a network. It characterizes and analyzes the data that flows between IP devices on the network to differentiate abnormal network behavior from normal network behavior [SW-03]. Unlike signature based IDS's, the appliance detects out-of-profile behaviors without state-fully inspecting packet traffic nor impeding volume throughput.

4. Anomaly Based IDS Functionality

The following describes the anomaly based IDS's basic functions and is taken from vendor literature.

a. Concern Index

Functionality is achieved through a patented "Concern Index" mechanism that is integrated within the appliance. The appliance measures the level of threatening activity occurring at the host level, and cumulatively, at the network level. When intrusive activity occurs on a host, the Concern Index (CI) accumulates points every time the activity occurs.

Each network host has an independent concern index threshold. As the concern index increases and exceeds the threshold, the appliance discretely notifies the administrator of the activity.

b. Target Index

The product used for this experiment also utilizes a "Target Index" that triggers when activity exceeds a pre-determined threshold. The alarm indicates that the target IP has received a number of probes or other malicious threats and has exceeded the set threshold [SW-03].

c. Behavior Profiling

- Host Profiling is the process of passively identifying and categorizing network resources. Acting as a sort of "passive port scanner," the appliance monitors network hosts' activity and builds a profile for each network host.
- Traffic Profiling monitors packet rate, bandwidth consumption, protocol usage, and traffic history statistics. Traffic profile thresholds are factored in to the flow based statistical analysis algorithms.

d. Flow-Based Statistical Analysis

The appliance uses a unique, patent pending flow-based packet capturing and analysis engine. As data-grams are received through the NIC's, they are fed into a flow analysis engine that separates and categories the active data flows.

Once these flows have been properly categorized, the appliance performs an analysis of the collected data, it checks both host and traffic profiles, and system-wide threshold settings to verify the flows satisfy the parameters of the established behavioral profile.

It is at this point that nefarious traffic is identified and reported. As patterns emerge and suspect flows are identified, the appliance accumulates CI points for the suspect host and alarms the administrator of mischievous activity.

Figure 7 illustrates the appliances "Triadic Threat Response" model.

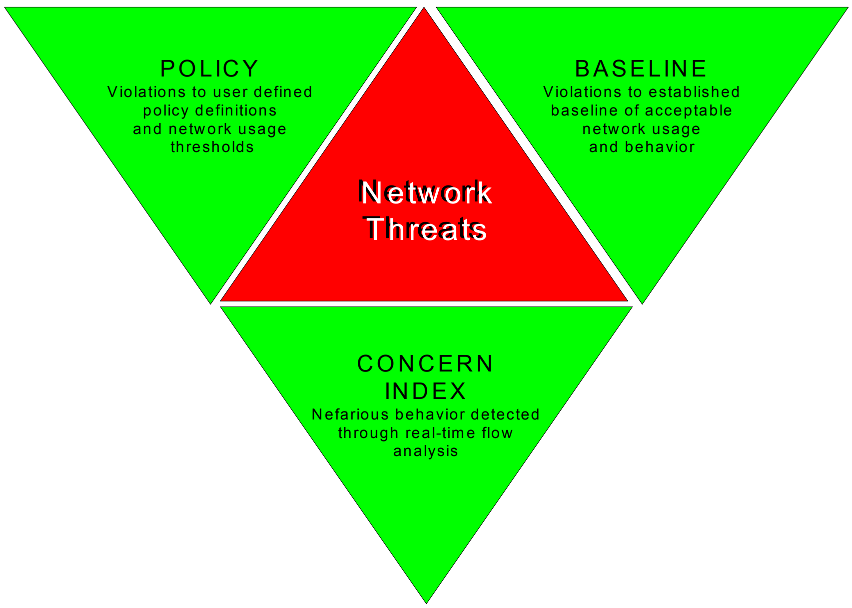


Figure 7. StealthWatch Triadic Threat Detection [From: SW-03]

D. QUANTITATIVE MEASURABLE IDS CHARACTERISTICS

In order to analyze all data collected properly during the experiment, it is prudent to identify metrics by which the data will be interpolated. They were chosen because they provide the ubiquitous characteristics of most IDS's on the commercial market today. Table 7 identifies these metrics.

Metric	Definition
Coverage	Determines which attacks an IDS can detect under ideal conditions. For SIGNATURE BASED IDS's, consists of counting the number of signatures and mapping them to a standard naming scheme.
Probability of False Alarms	Determines the rate of false positives produced by an IDS in a given environment during a particular time frame. A false positive is an alert caused by normal non-malicious background traffic.
Probability of Detection	Determines rate of attacks detected correctly by an IDS in a given environment during a particular time frame.
Resistance to Attacks Directed at the IDS	Demonstrates IDS resistance is to an attacker's attempt to disrupt the correct operation of the IDS.
Ability to Handle High Bandwidth Traffic	Demonstrates how well an IDS functions when presented with a large volume of traffic.
Ability to Correlate Events	Demonstrates how well an IDS correlates attack events. They may be gathered from IDS's, routers, firewalls, application logs, or other devices.
Ability to Detect Zero Day Attacks	Demonstrates how well an IDS detects attacks that have not occurred before. Anomaly based systems are better suited for this type of measurement.
Ability to Identify an Attack	Demonstrates how well an IDS can identify the detected attack by labeling each attack with a common name or vulnerability name or by assigning it a category.
Ability to Determine Attack Success	Demonstrates if the IDS can determine the success of attacks from remote sites that give the attacker escalated privileges on the compromised system.
Capacity Verification for NIDS	The NIDS demands higher-level protocol awareness than other network devices such as switches and routers; it has the ability of inspection into the deeper level of network packets.

Table 7. Quantitative Measurable Characteristics of IDS's [From: NIST-03]

E. CHAPTER SUMMARY

This chapter detailed the test bed hardware and software configuration of the STOOGE-CENTRAL network. It depicted the setup of the two enclave boundary defenses that will be used to capture, monitor, and report all network traffic that traverses this network. Furthermore, it described in detail functionalities associated with these two network based IDS's. The chapter concluded by introducing the metrics that will be utilized to analyze and interpolate the captured network traffic.

IV. TEST DATA RESULTS AND EVALUATION

Anyone who has never made a mistake has never tried anything new.

Albert Einstein

A. PROLOGUE

This chapter will discuss and describe the data normalization process and the data collected during the IDS experiment. It will first detail all identified mischievous traffic collected during the experiment. It will then discuss matters relevant to the signature based IDS followed by a discussion of those occurrences relevant to the anomaly based IDS. To conclude the chapter, an IDS cumulative comparison analysis involving line plots, regression analysis, and significance testing will occur.

B. DATA NORMALIZATION PROCESS

1. Collection Period

The IDS experiment collection date began 1 May 2004 and concluded 19 June 2004. This 50 day window allowed for the monitoring of traffic flow and the capture of traffic for a contiguous hourly timeframe. At the experiment's conclusion, reports were generated from both IDS appliances and subsequently imported into Microsoft's Excel and Access programs. Excel was chosen for use as the statistical analysis engine and Access for the data knowledge base.

2. Identified Mischievous Occurrences

The total mischievous traffic collected over the inclusive period was 7672 conversations with the signature based IDS capturing 4085 and the anomaly based IDS reporting 3587. Since it is the Information Assurance (IA) professional's belief that anomaly based IDS's incur more "false positives" than signature based systems, these "false positives" must be excluded from those identified mischievous occurrences. False Positives are defined as "a positive result when in reality it is negative in nature" [WEB-03]. Therefore, in order to rule out most false positives both reports were juxtaposed and a line-by-line comparison of Internet Protocol (IP) addresses and date-time-groups revealed 1412 common nefarious occurrences. Figure 8 is a Venn diagram that depicts this information.

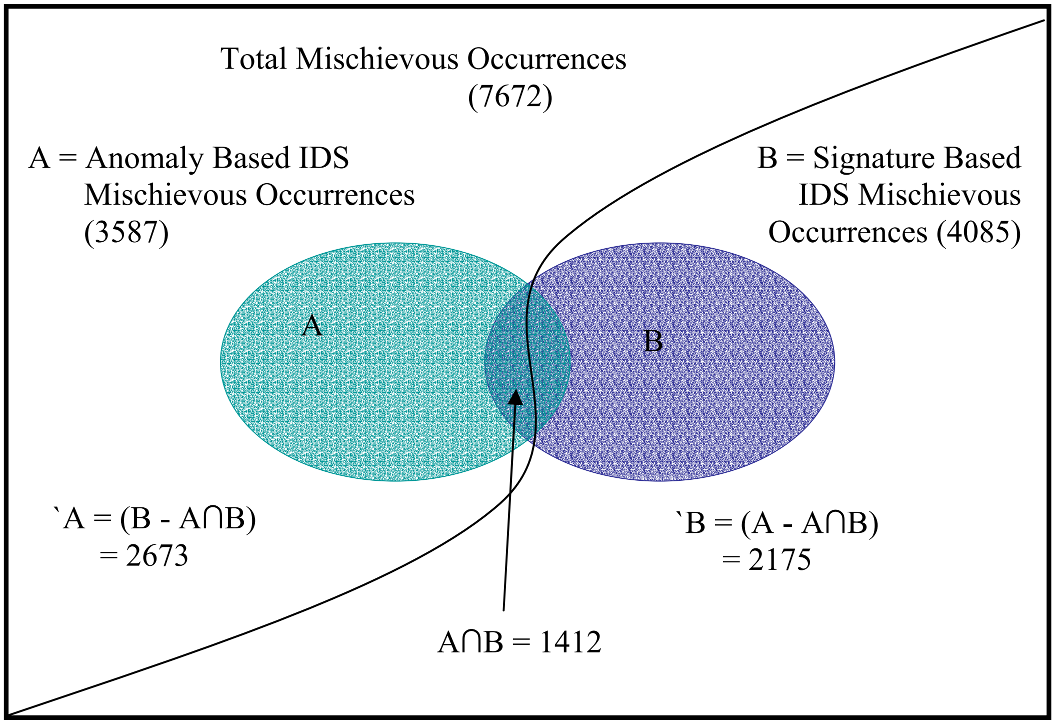


Figure 8. Total Mischievous Occurrence Diagram

These common occurrences were then inspected for any STOOGE-CENTRAL internal LAN traffic, Yahoo CHAT "pings" and "logons," conversations between the NPS 131.120.255.255 subnet that were not port 135 or port 445 virus traffic, and Hotmail e-mail traffic exchanges. If identified as listed above, these occurrences were ruled out as "false positives" and subsequently subtracted from the 1412. This reduced the mischievous common traffic of both the signature based and anomaly based IDS's to 1333 and 1369, respectively.

3. Normalizing Data

The method used to normalize the mischievous common traffic was first to import all data into Microsoft's Excel. In order to add consistency throughout the analyzing process, it was determined that comparing the number of mischievous common occurrences to the time of day would be the most appropriate measure. Therefore, a spreadsheet was used to determine each occurrence's date and time. It looked at the time and rounded either up or down to the "whole hour" based on the "half-hour" system. If the time was greater than or equal to 30, the time was rounded up to the next whole hour, if it was less than 30, then it was rounded back to the original hour.

This process provided the ability to categorize mischievous traffic into three groups of eight-hour time blocks. All mischievous common traffic was then analyzed in time blocks that ranged from 0100 – 0800, 0900 – 1600, 1700 – 2400, and a 24-hour cumulative time block. Once sub-divided into these three groups, a statistical analysis of this traffic began.

C. SIGNATURE BASED IDS DATA DESCRIPTION

The signature based IDS collected 53% of the total wayward traffic that traversed the STOOGE-CENTRAL network. This amount includes both identified mischievous traffic and all false positives. In order to assess the collected data fairly, it must be scrubbed for accuracy and all false positives removed. Therefore, after excluding those considered false, 1333 conversations were considered as "roguish," which equates to 17% of the total amount of data collected. Figure 9 graphically depicts this.

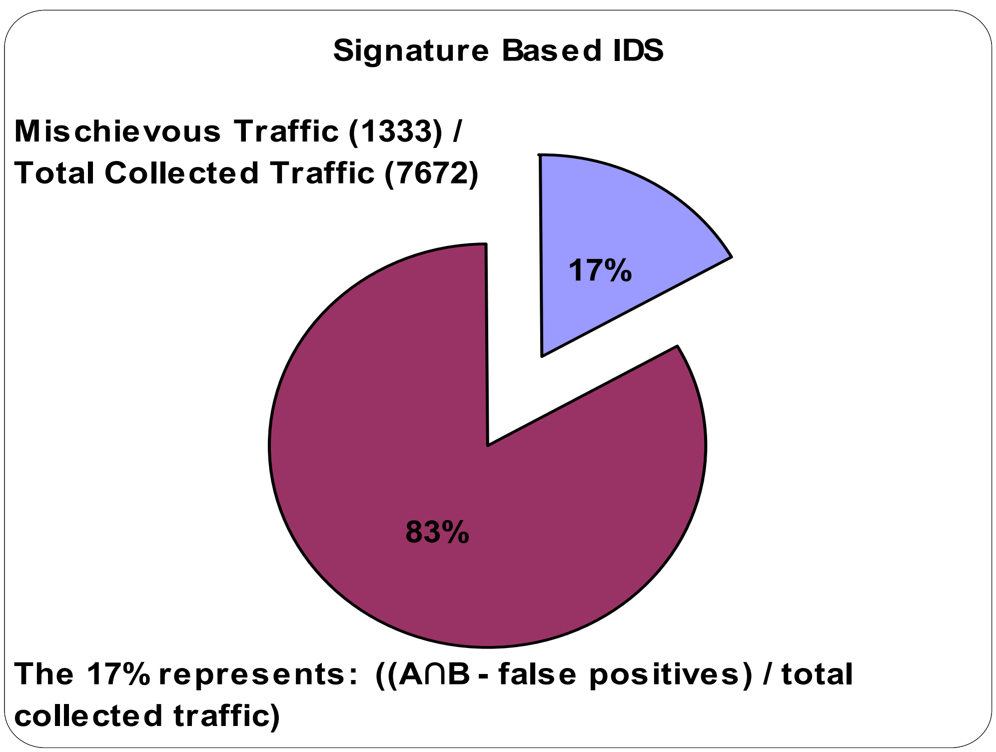


Figure 9. Signature Based IDS Harmful Traffic

Analysis of this traffic reveals that in the 50 day window most occurred during the 1700 – 2400 timeframe. Figure 10, below, reveals this and the signature based IDS Pie Chart found in Figure 11 shows each number of occurrence's percentage.

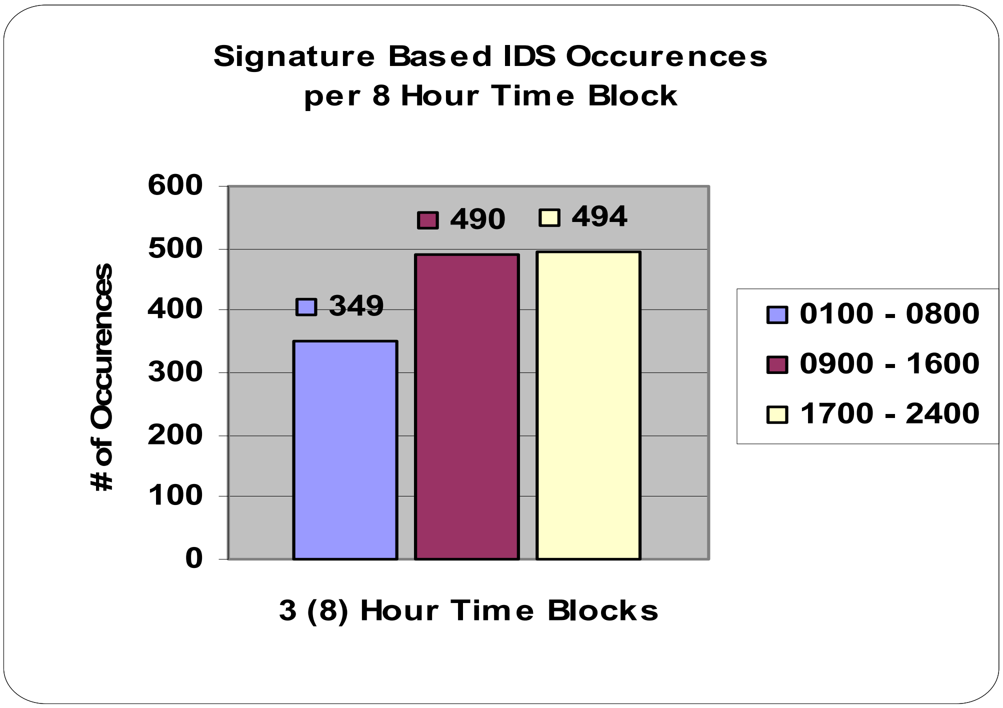


Figure 10. Signature Based IDS Bar Graph

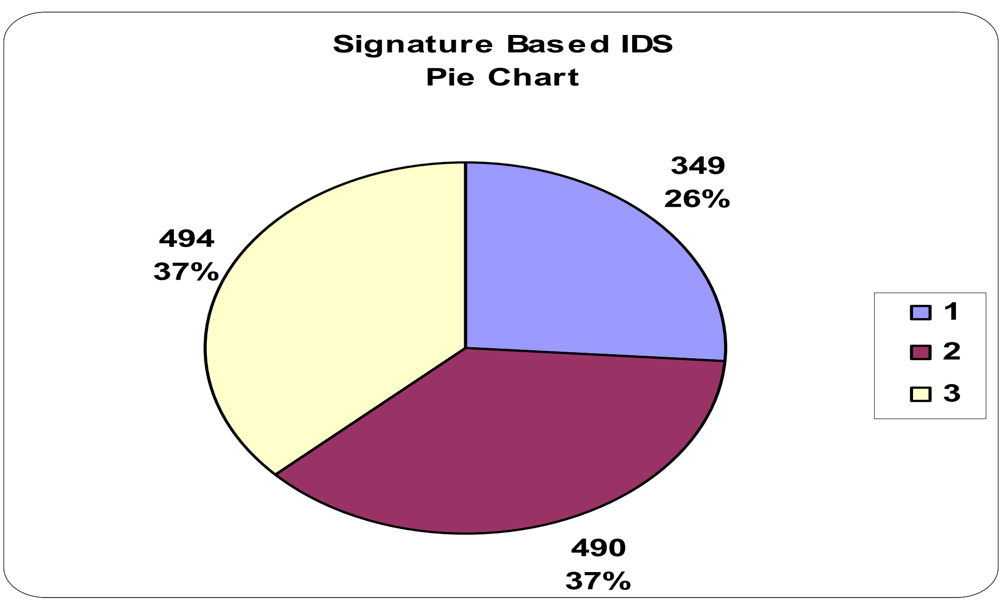


Figure 11. Signature Based IDS Pie Chart

1. 0100 - 0800 Time Frame Analysis

Table 8, below, represents this time frame with the number of total mischievous occurrences per rounded hour. It also includes the Standard Deviation (σ), Mean (μ), Median, and Mode for this period. Figure 12 portrays the number of roguish occurrences for each hour in a bar graph.

0100 - 0800 Time Period		Standard Deviation
Rounded Hour	**Total**	14.17
1	62	
2	36	Mean
3	34	43.63
4	24	
5	31	Median
6	49	42.5
7	59	
8	54	Mode
Grand Total	349	#N/A

Table 8. Signature Based IDS 0100 – 0800 Time Frame

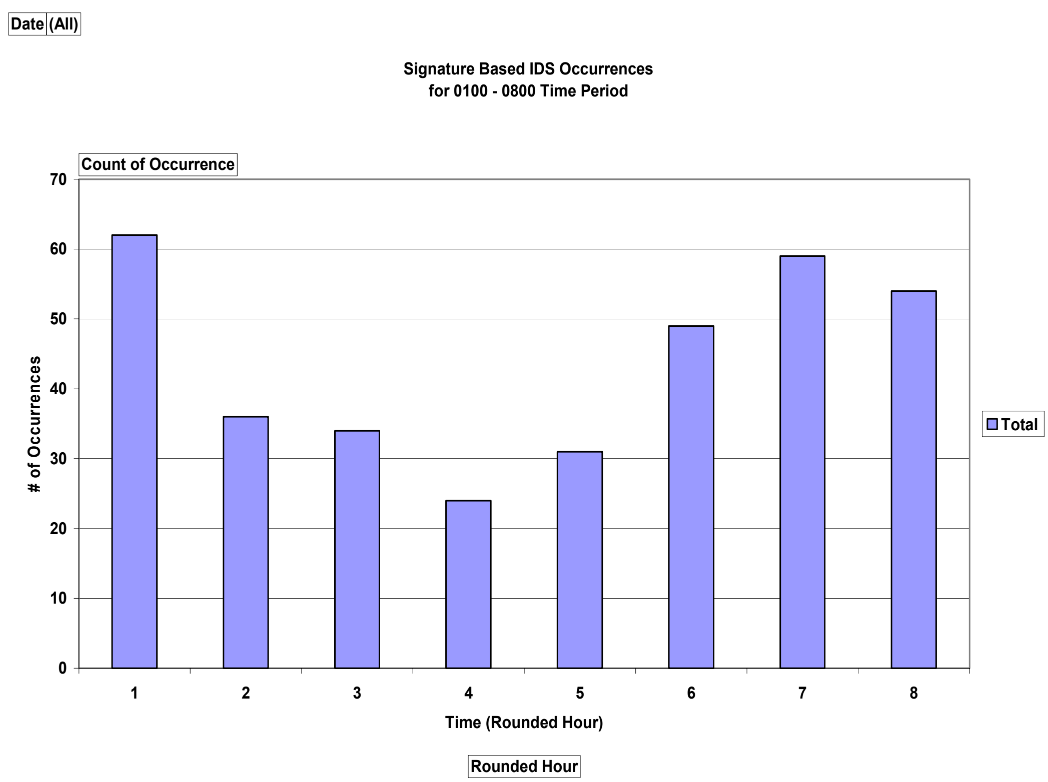


Figure 12. Signature Based IDS 0100 – 0800 Bar Graph

When analyzing the standard deviation for this period and all subsequent periods for both the signature based and anomaly based IDS's mischievous traffic, application of the "68-95-99.7" rule applies [MOORE-01]. This rule states that "in the normal distribution with mean μ and standard deviation σ:"

- 68% of the observations fall within σ of the mean μ.
- 95% of the observations fall within 2σ of μ.
- 99.7% of the observations fall within 3σ of μ.

A plot of this time period's total roguish occurrences per rounded hour reveals that only 62.5% of occurrences fall within σ of μ. Although this number falls outside the 68% rule and is not significantly high, it must be mentally noted.

Figure 13 is a Normal Distribution (Bell) Curve that represents Table 8's statistical data.

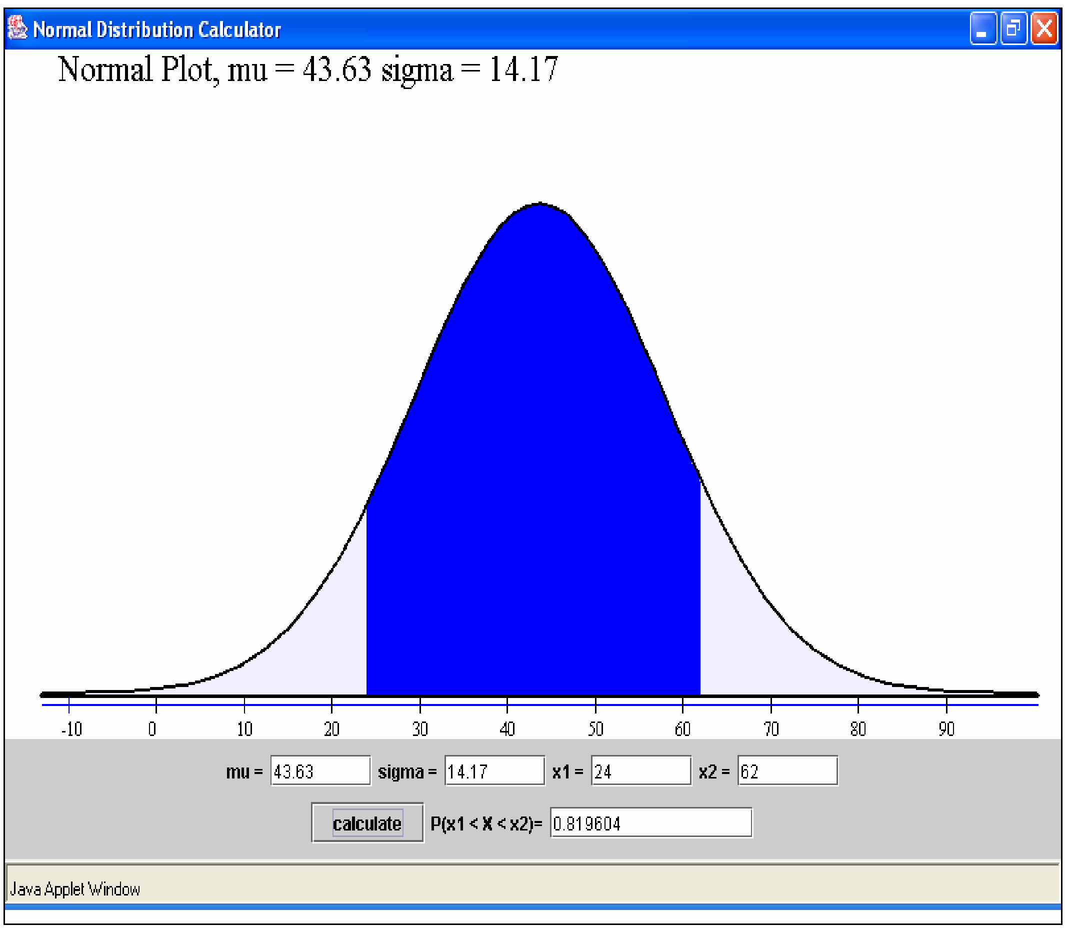


Figure 13. Signature Based IDS 0100 – 0800 Distribution Curve [From: CSUSB-04]

Figure 14, below, represents a line plot of the total mischievous occurrences collected during the 50 day test date range specifically for this time period.

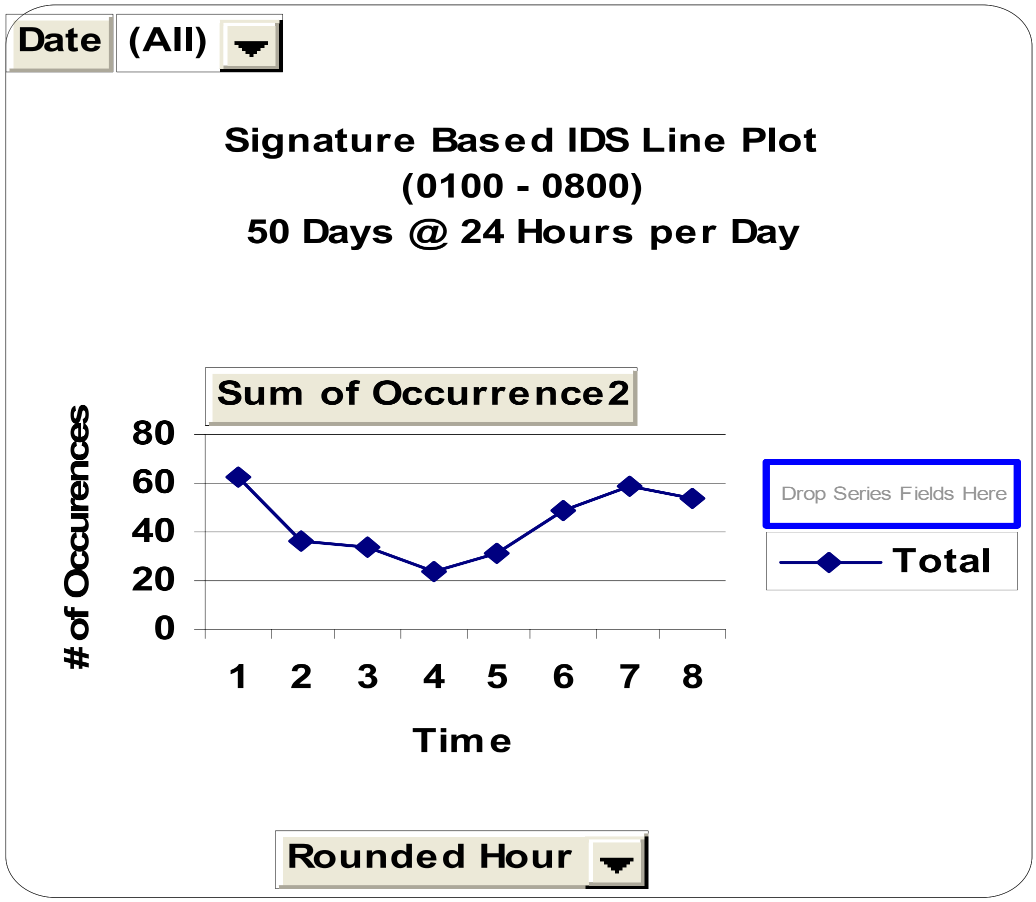


Figure 14. Signature Based IDS 0100 – 0800 Line Plot

2. 0900 – 1600 Time Frame Analysis

Table 9, below, represents this time frame with the number of mischievous occurrences per rounded hour. Figure 15 portrays the number of occurrences for each hour in a bar graph.

0900 - 1600 Time Period		Standard Deviation
Rounded Hour	Total	18.05
9	95	
10	48	Mean
11	48	61.25
12	48	
13	42	Median
14	65	56.5
15	72	
16	72	Mode
Grand Total	490	48

Table 9. Signature Based IDS 0900 – 1600 Time Frame

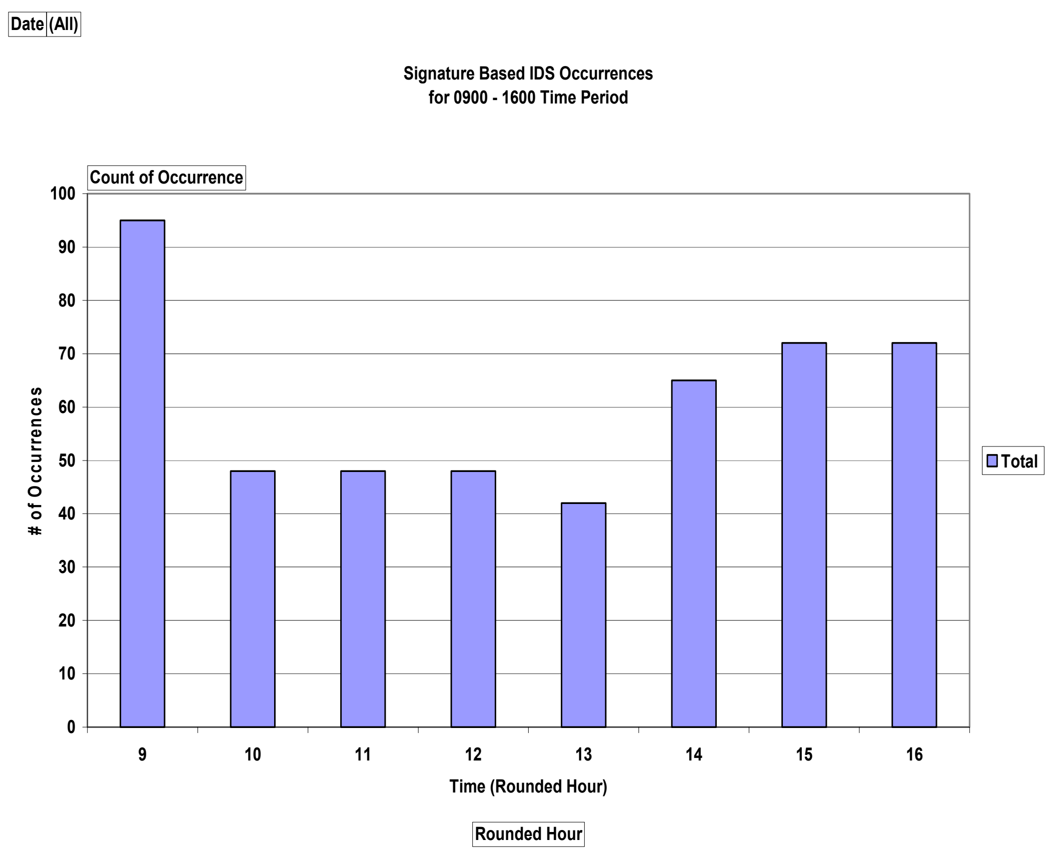


Figure 15. Signature Based IDS 0900 – 1600 Bar Graph

A plot of these occurrences per rounded hour reveals that, for this time period, 75% of occurrences fall within σ of μ. This number falls well within the 68% rule. Figure 16 represents Table 9's statistical data.

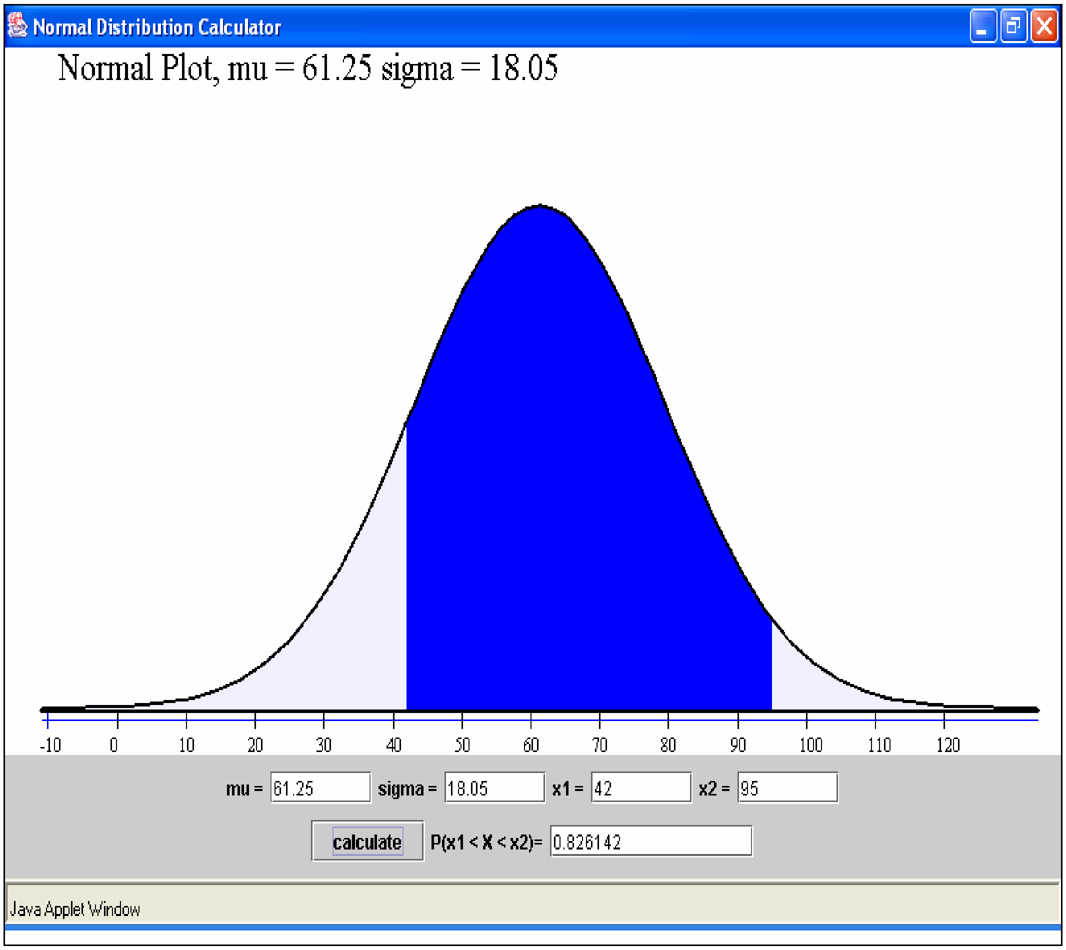


Figure 16. Signature Based IDS 0900 – 1600 Distribution Curve [From: CSUSB-04]

Figure 17, below, represents a line plot of these mischievous occurrences collected during the 50 day test date range specifically for this time period.

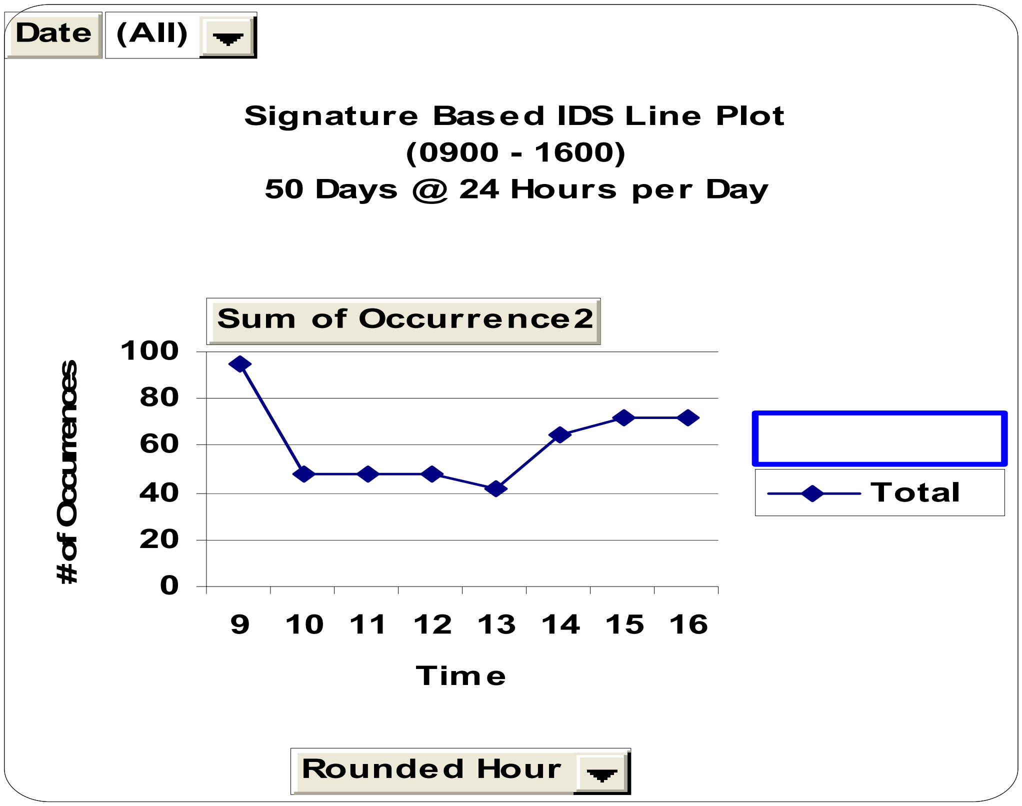


Figure 17. Signature Based IDS 0900 – 1600 Line Plot

3. 1700 – 2400 Time Frame Analysis

Table 10, below, represents this time frame with the total mischievous occurrences per rounded hour. Figure 18 represents this in a bar graph.

1700 - 2400 Time Period		Standard Deviation
Rounded Hour	Total	20.59
17	71	
18	62	Mean
19	64	61.75
20	91	
21	41	Median
22	37	63.00
23	86	
24	42	Mode
Grand Total	494	#N/A

Table 10. Signature Based IDS 1700 – 2400 Time Frame

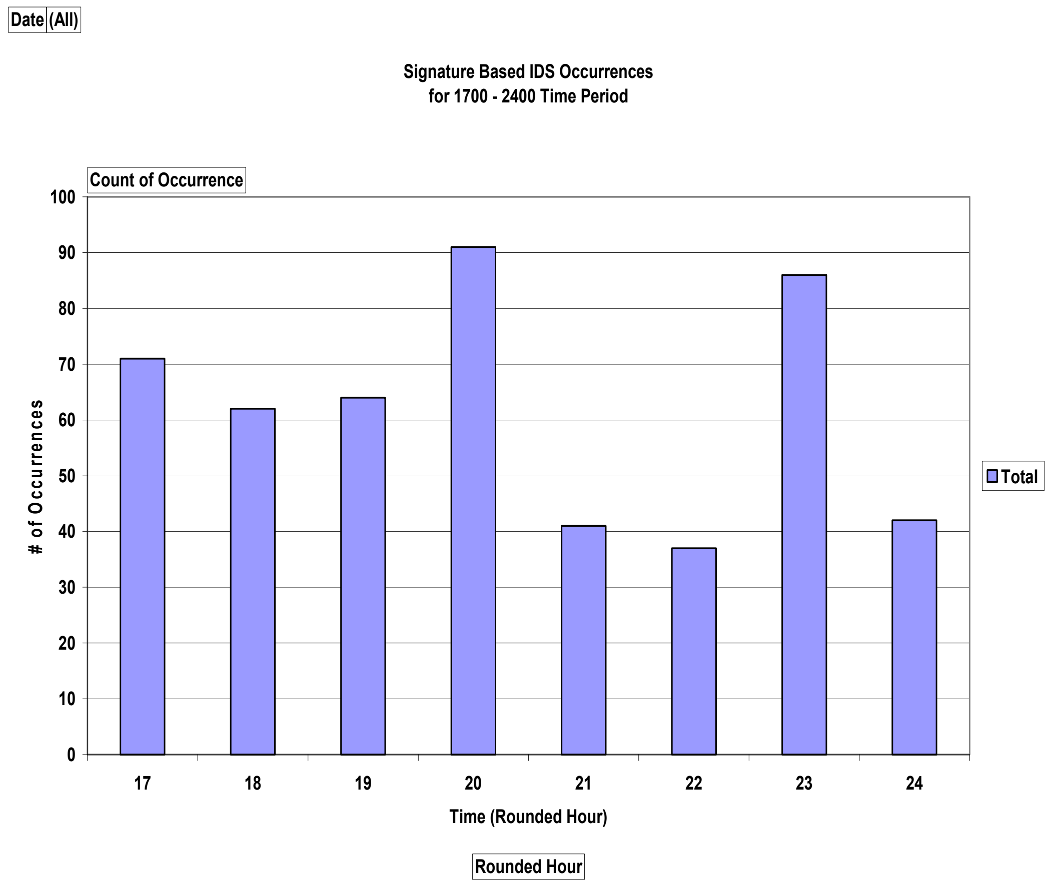


Figure 18. Signature Based IDS 1700 – 2400 Bar Graph

A plot of these occurrences per rounded hour reveals that, for this time period, only 50% of occurrences fall within σ of μ. Although this number falls well outside the 68% rule, it too must be kept in mind.

Figure 19 represents Table 10's statistical data.

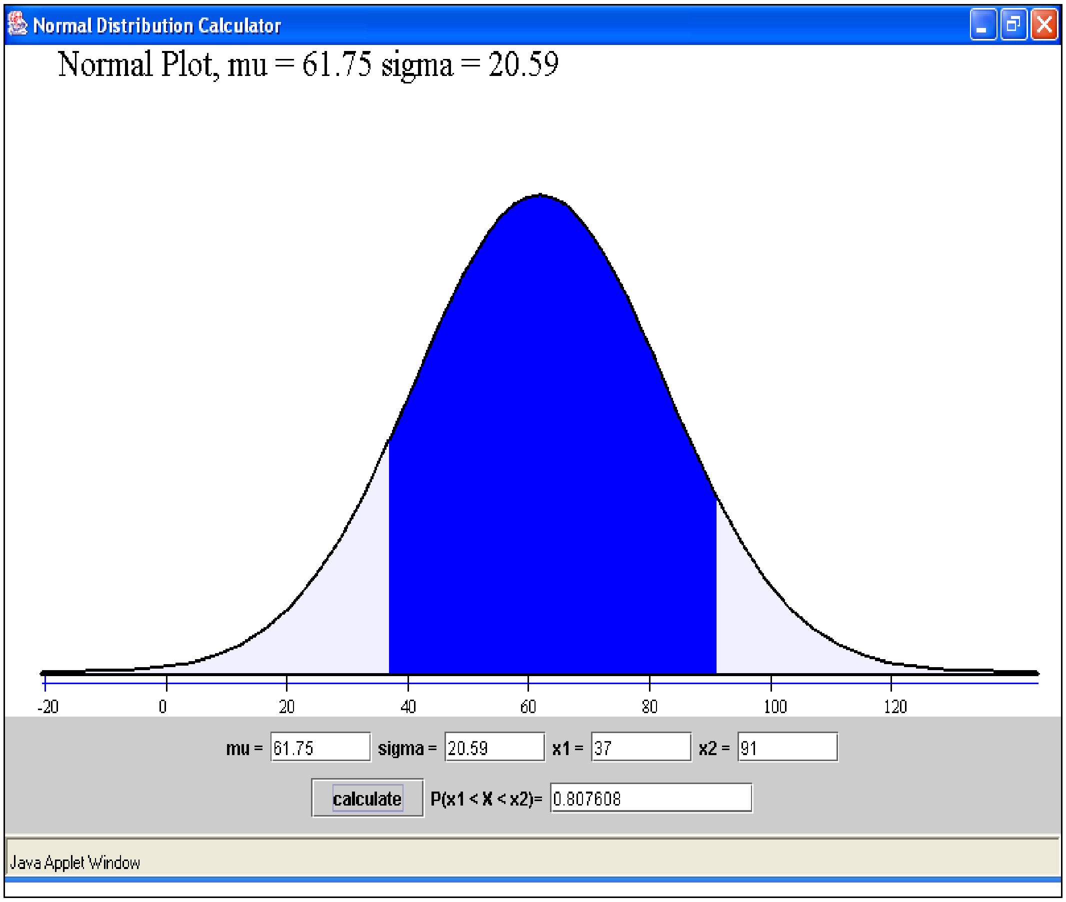


Figure 19. Signature Based IDS 1700 – 2400 Distribution Curve [From: CSUSB-04]

Figure 20, below, represents a line plot of these mischievous occurrences collected during the 50 day test date range specifically for this time period.

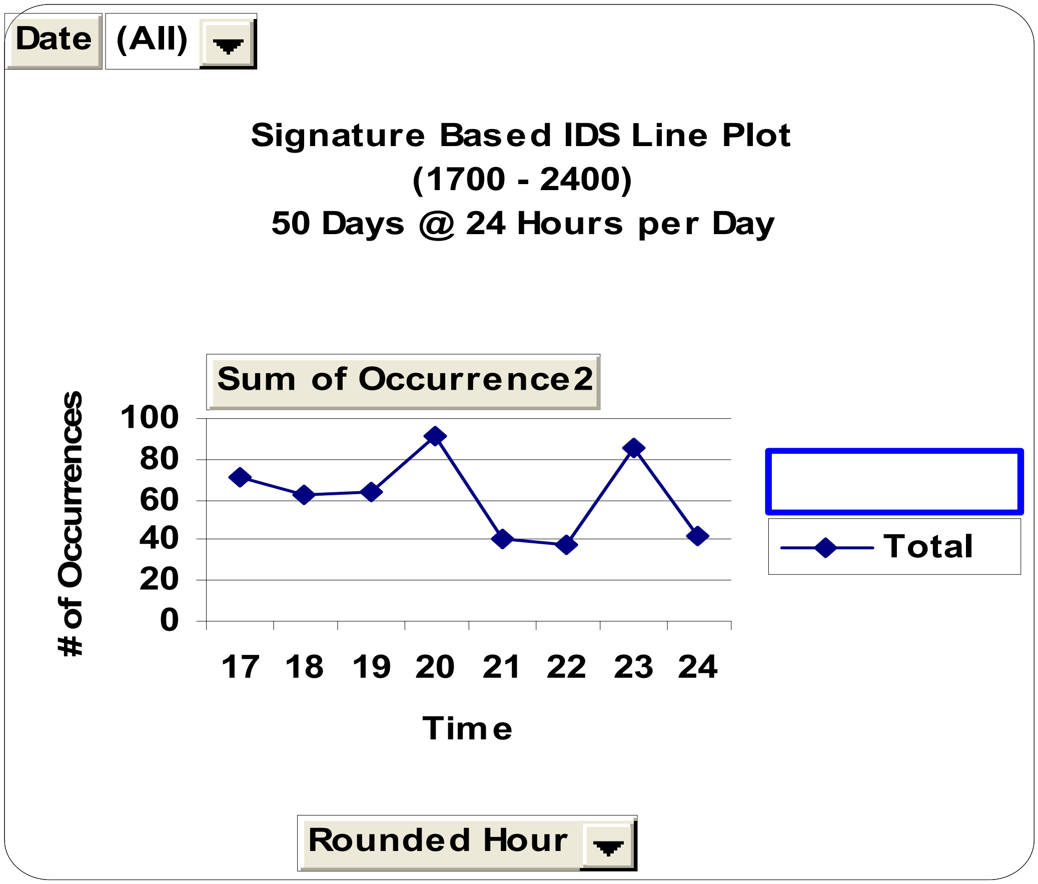


Figure 20. Signature Based IDS 1700 – 2400 Line Plot

4. Signature Based IDS Cumulative 24 Hour Period

Table 11 below, represents this cumulative period with the total mischievous occurrences per rounded hour. Figure 21 represents this in a bar graph.

24 Hour Time Period		Standard Deviation
Rounded Hour	Total	19.07
1	62	
2	36	Mean
3	34	55.54
4	24	
5	31	Median
6	49	51.50
7	59	
8	54	Mode
9	95	48
10	48	
11	48	Range

12	48	71
13	42	
14	65	Minimum
15	72	24
16	72	
17	71	Maximum
18	62	95
19	64	
20	91	Standard Error
21	41	3.89
22	37	
23	86	Confidence Level (95.0%)
24	42	8.05
Grand Total	1333	

Table 11. Signature Based IDS Cumulative 24 Hour Period

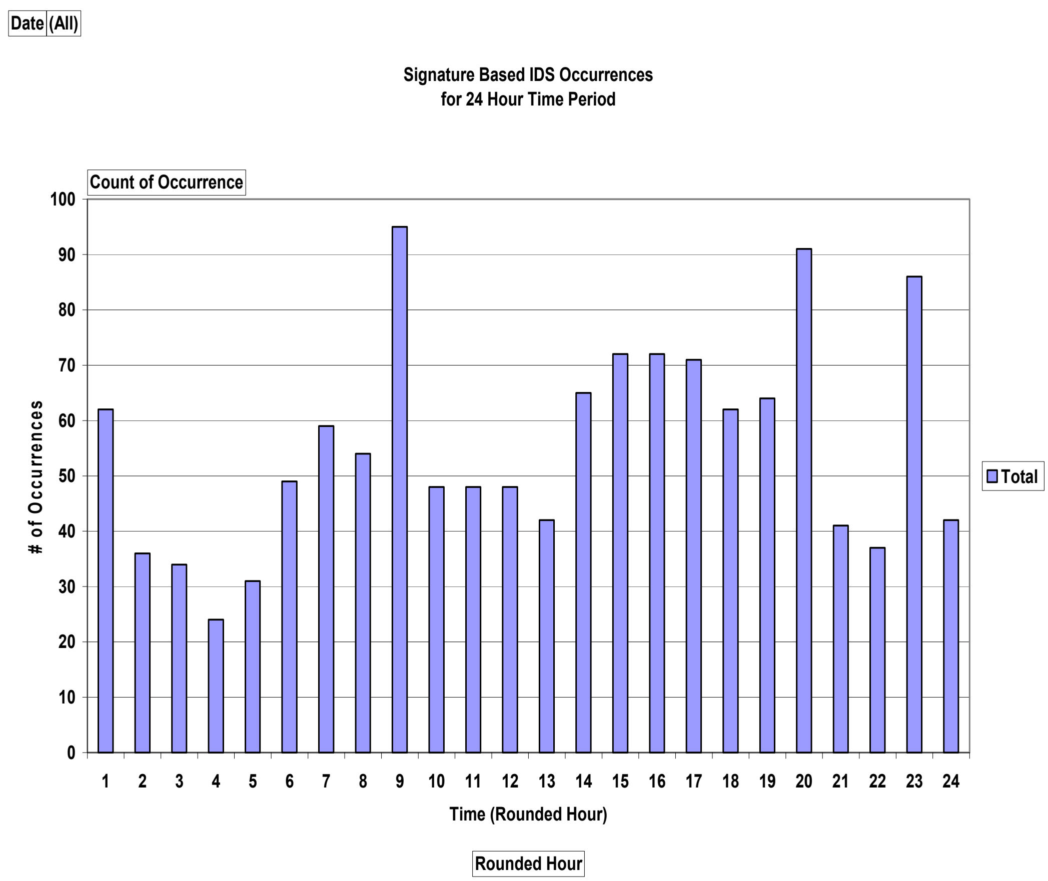


Figure 21. Signature Based IDS Cumulative Bar Graph

A plot of these occurrences per cumulative time frame reveals that, for this time period, 70.8% of occurrences fall within σ of μ. This high percentage makes it tempting to assume that the signature based IDS proves the Null Hypothesis as statistically significant; however, further testing must occur.

Figure 22 represents Table 11's statistical data.

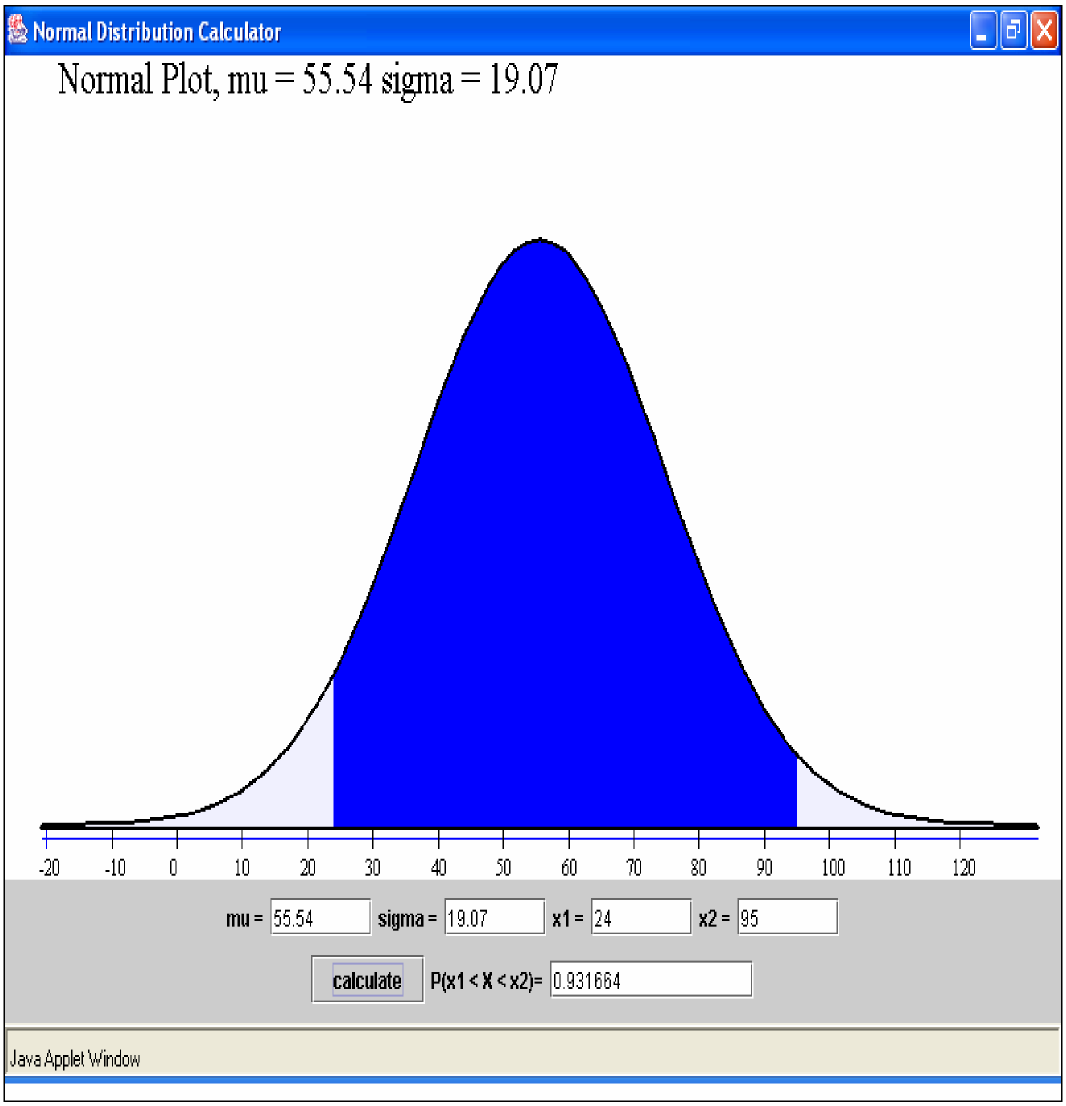


Figure 22. Signature Based IDS Cumulative Distribution Curve [From: CSUSB-04]

Figure 23, below, represents a line plot of these mischievous occurrences collected during the 50 day test date range specifically for this time period.

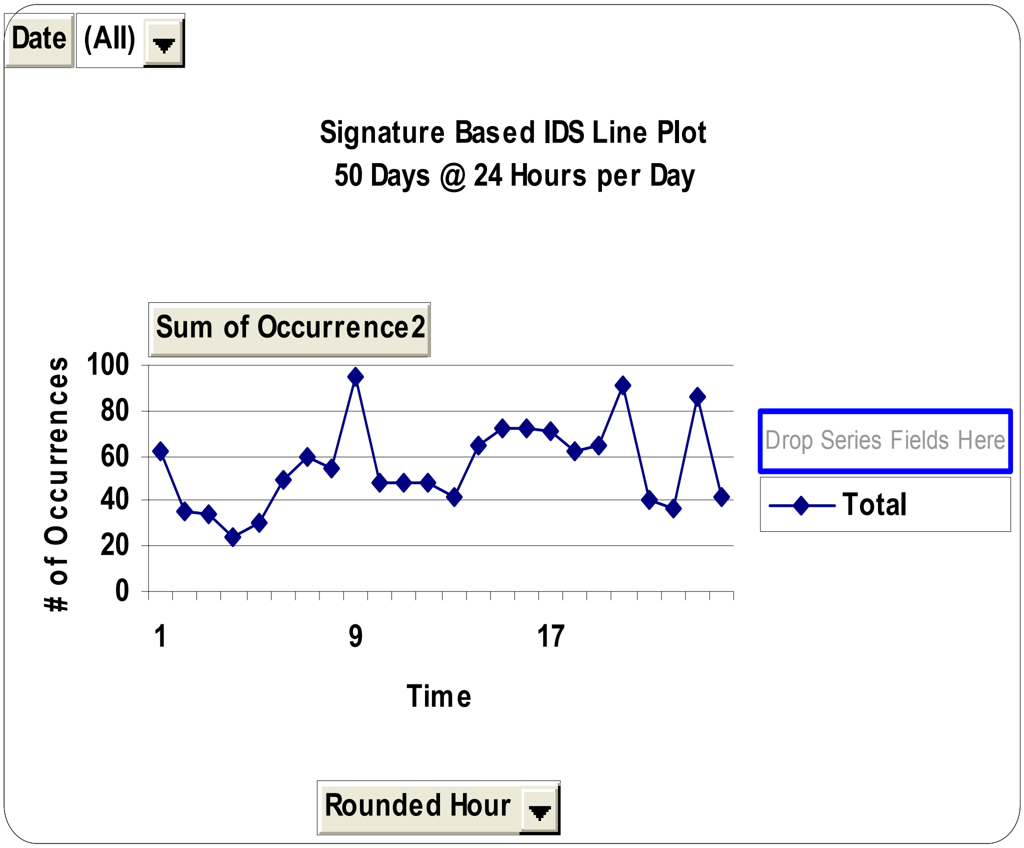


Figure 23. Signature Based IDS Cumulative 24 Hour Period Line Plot

The preceding figure reveals that although, as previously noted in Figure 10, most wayward occurrences were *recorded* in the 1700 - 2400 time block, the actual time with the highest amount of nefarious traffic is 0900.

D. ANOMALY BASED IDS DATA DESCRIPTION

The anomaly based IDS collected 47% of the total mischievous traffic that traversed the STOOGE-CENTRAL network. This total amount includes both mischievous traffic and all false positives. In order to assess the collected data fairly, it must be scrubbed for accuracy, which means removing all false positives. Therefore, after excluding those considered false, 1369 conversations were considered as "roguish," which equates to 18% of the total amount of data collected. Figure 24 graphically depicts this.

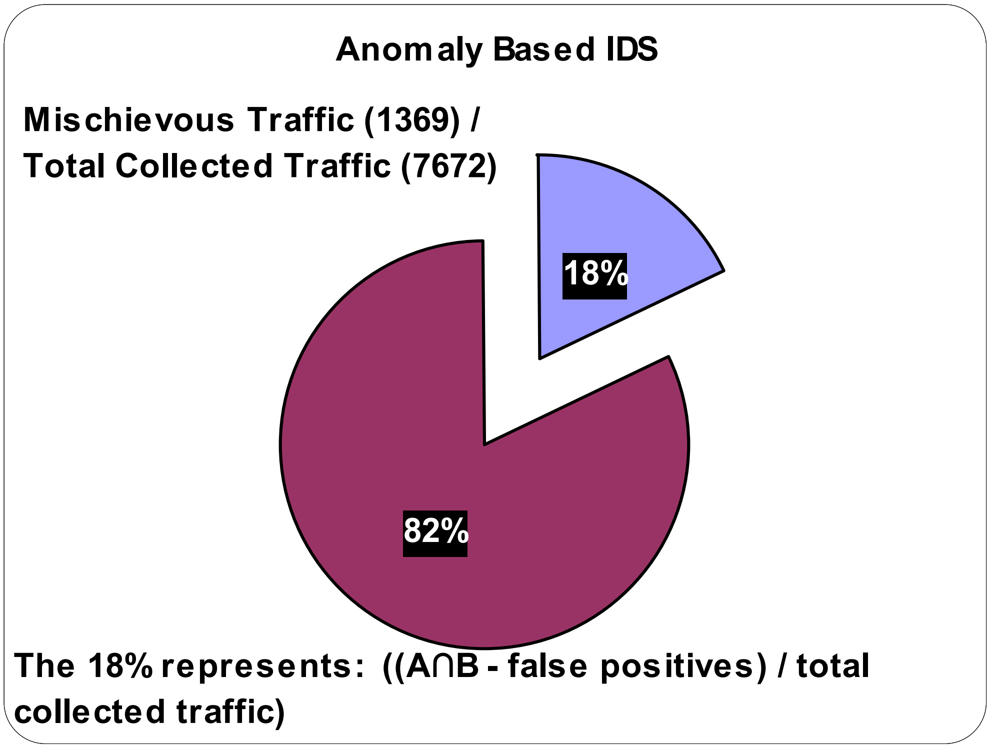


Figure 24. Anomaly Based IDS Harmful Traffic

Analysis of the mischievous traffic reveals that in the 50 day window most traffic collected for the anomaly based IDS occurred during the 0900 – 1600 timeframe. Figure 25 reveals this and the anomaly based IDS pie chart found in Figure 26 shows each occurrence's percentage.

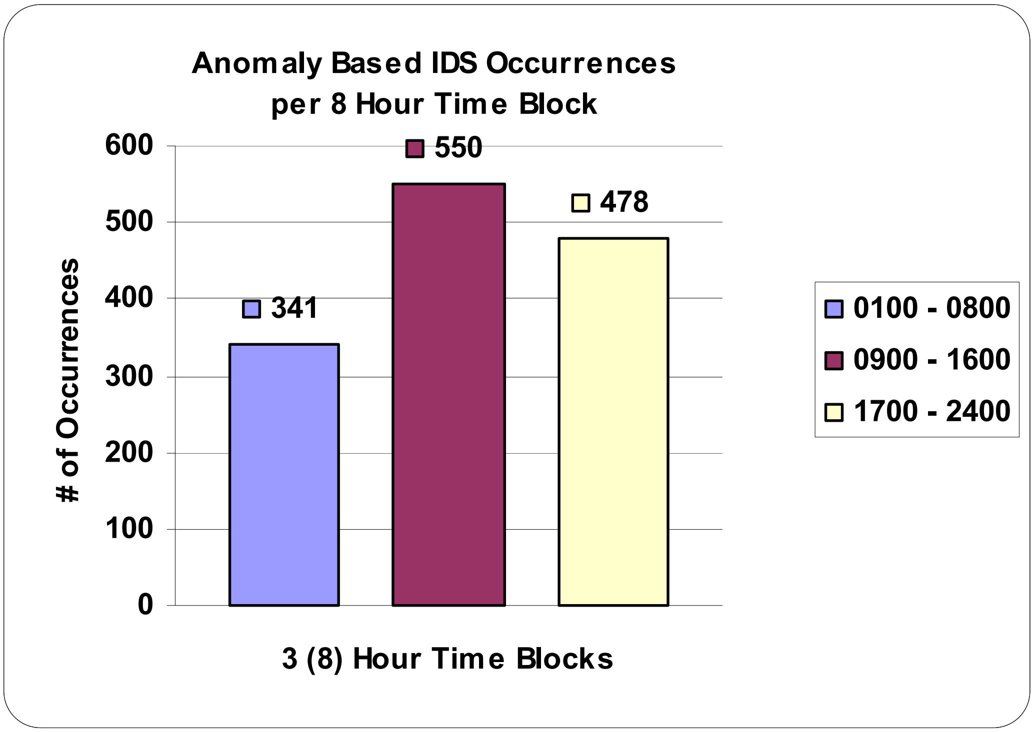


Figure 25. Anomaly Based IDS Bar Graph

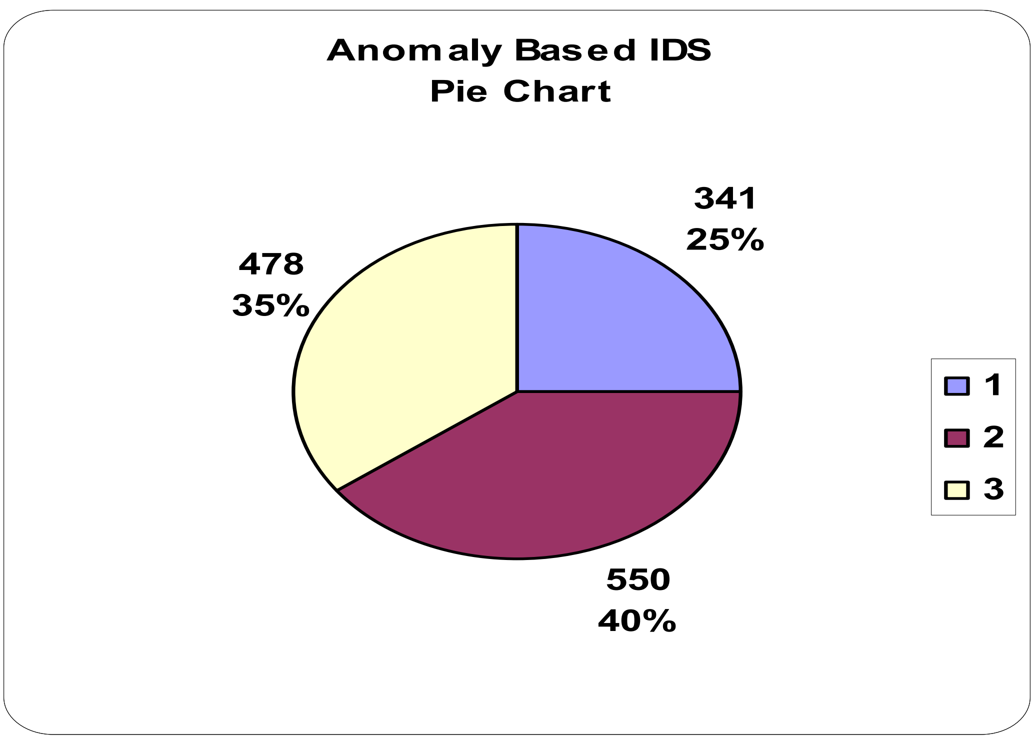


Figure 26. Anomaly Based IDS Pie Chart

1. 0100 - 0800 Time Frame Analysis

Table 12, below, represents this time frame with the number of mischievous occurrences per rounded hour. It also includes the Standard Deviation (σ), Mean (μ), Median, and Mode for this period. Figure 27 portrays these occurrences for each hour in a bar graph.

0100 - 0800 Time Period		Standard Deviation
Rounded Hour	Total	12.89
1	63	
2	42	Mean
3	30	42.63
4	26	
5	33	median
6	41	41.5
7	52	
8	54	mode
Grand Total	341	#N/A

Table 12. Anomaly Based IDS 0100 – 0800 Time Frame

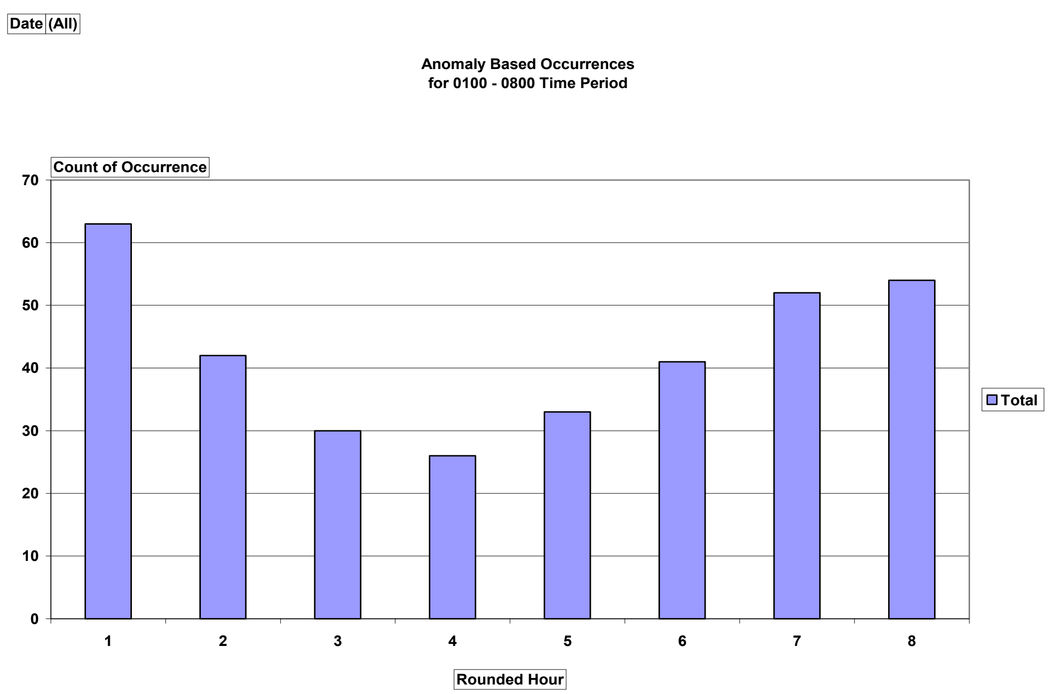


Figure 27. Anomaly Based IDS 0100 – 0800 Bar Graph

A plot of these occurrences per rounded hour reveals that, for this time period, 75% of occurrences fall within σ of μ.

Figure 28 represents Table 12's statistical data.

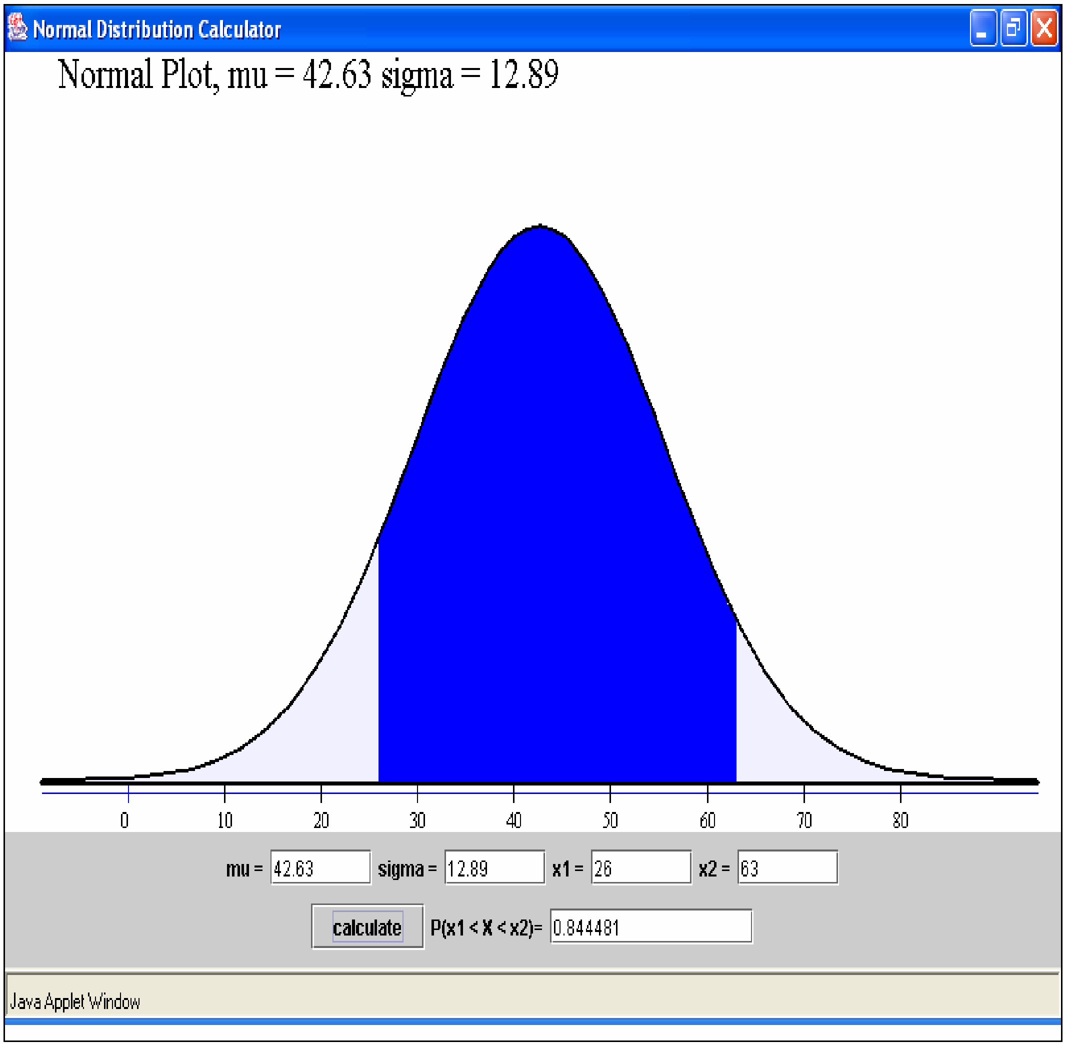


Figure 28. Anomaly Based IDS 0100 – 0800 Distribution Curve [From: CSUSB-04]

Figure 29, below, represents a line plot of these mischievous occurrences collected during the 50 day test date range specifically for this time period.

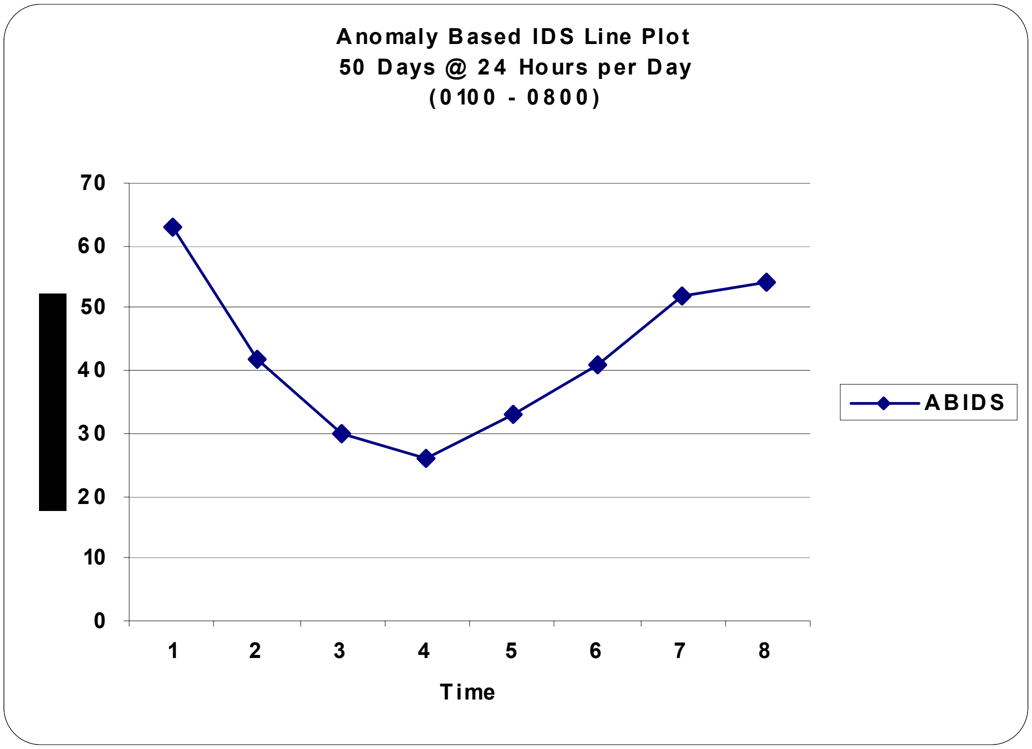


Figure 29. Anomaly Based IDS 0100 – 0800 Line Plot

2. 0900 – 1600 Time Frame Analysis

Table 13, below, represents this time frame with the number of total roguish occurrences per rounded hour. Figure 30 portrays these occurrences for each hour in a bar graph.

0900 - 1600 Time Period		Standard Deviation
Rounded Hour	Total	15.45
9	102	
10	69	Mean
11	51	68.75
12	62	
13	55	Median
14	68	68.5
15	71	
16	72	Mode
Grand Total	550	#N/A

Table 13. Anomaly Based IDS 0900 – 1600 Time Frame

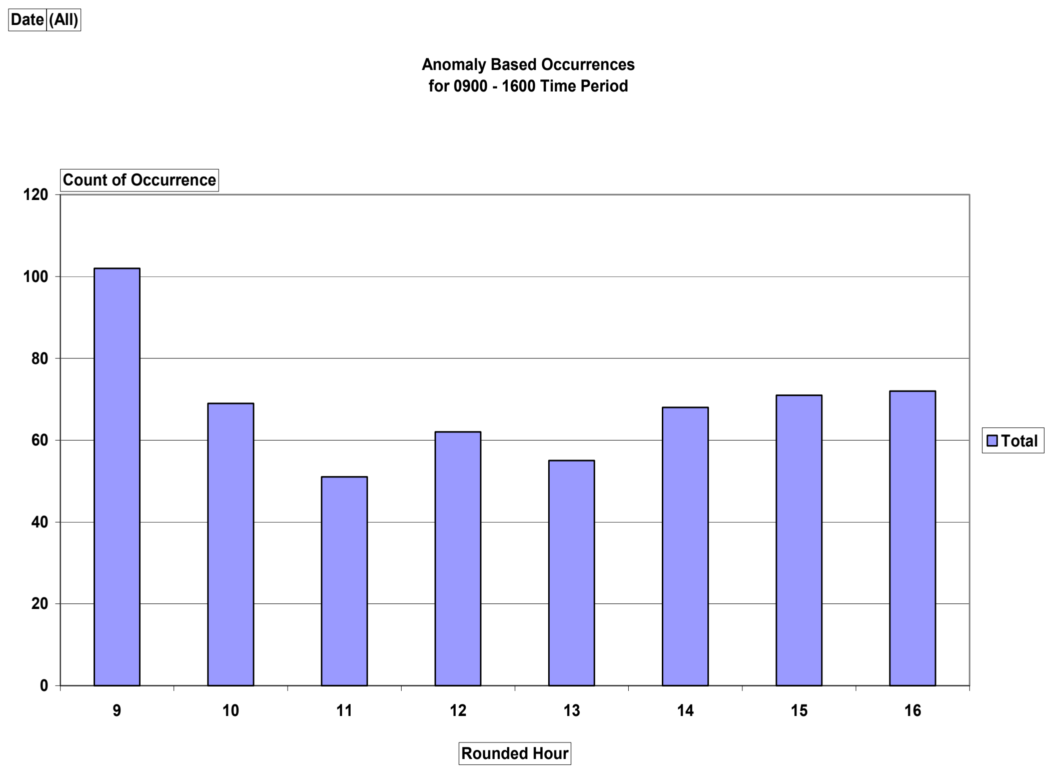


Figure 30. Anomaly Based IDS 0900 – 1600 Bar Graph

A plot of these occurrences per rounded hour reveals that, for this time period, 87.5% of occurrences fall within σ of μ. This number falls well within the 68% rule. Figure 31 represents Table 13's statistical data.

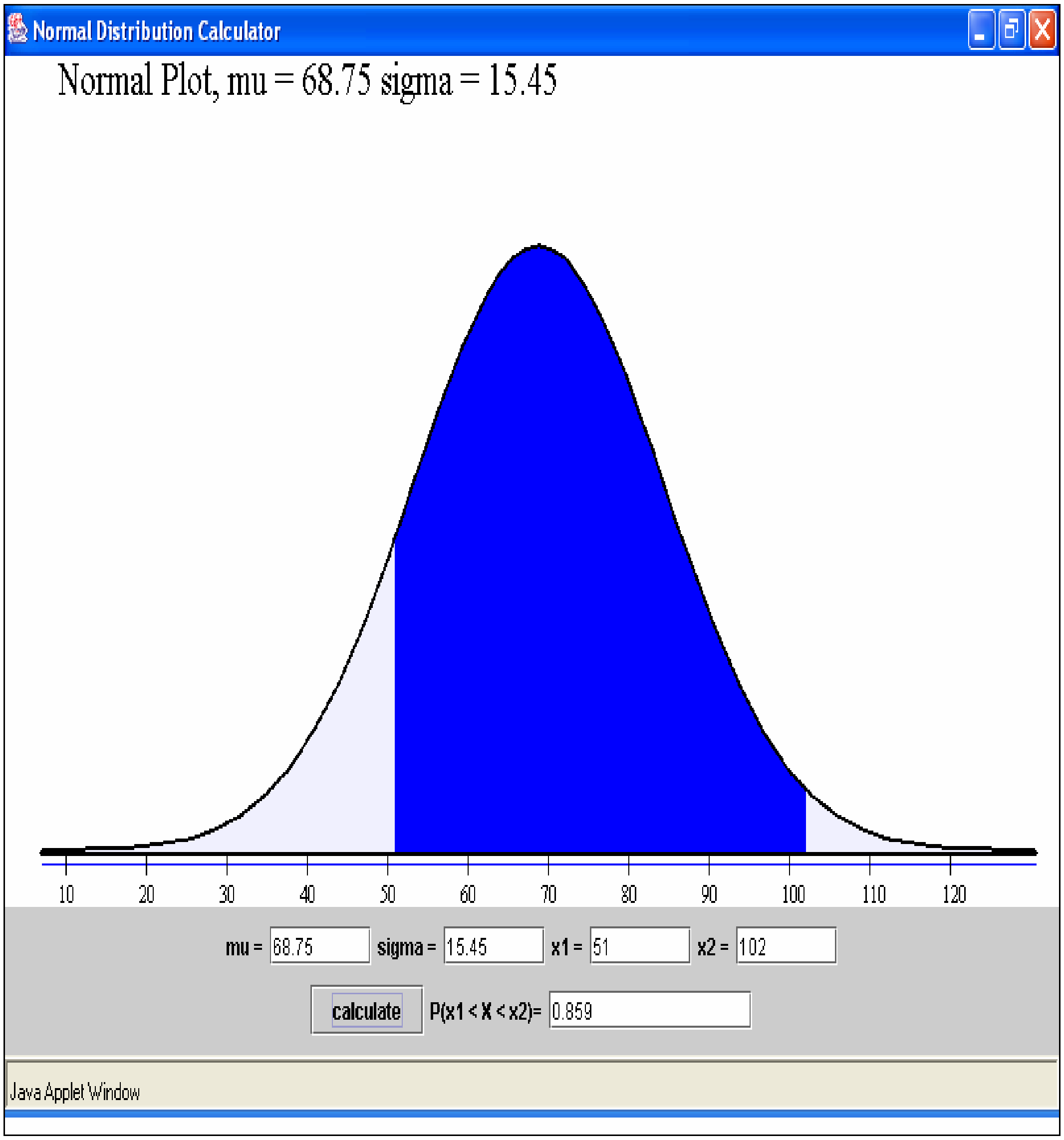


Figure 31. Anomaly Based IDS 0900 – 1600 Distribution Curve [From: CSUSB-04]

Figure 32, below, represents a line plot of these mischievous occurrences collected during the 50 day test date range specifically for this time period.

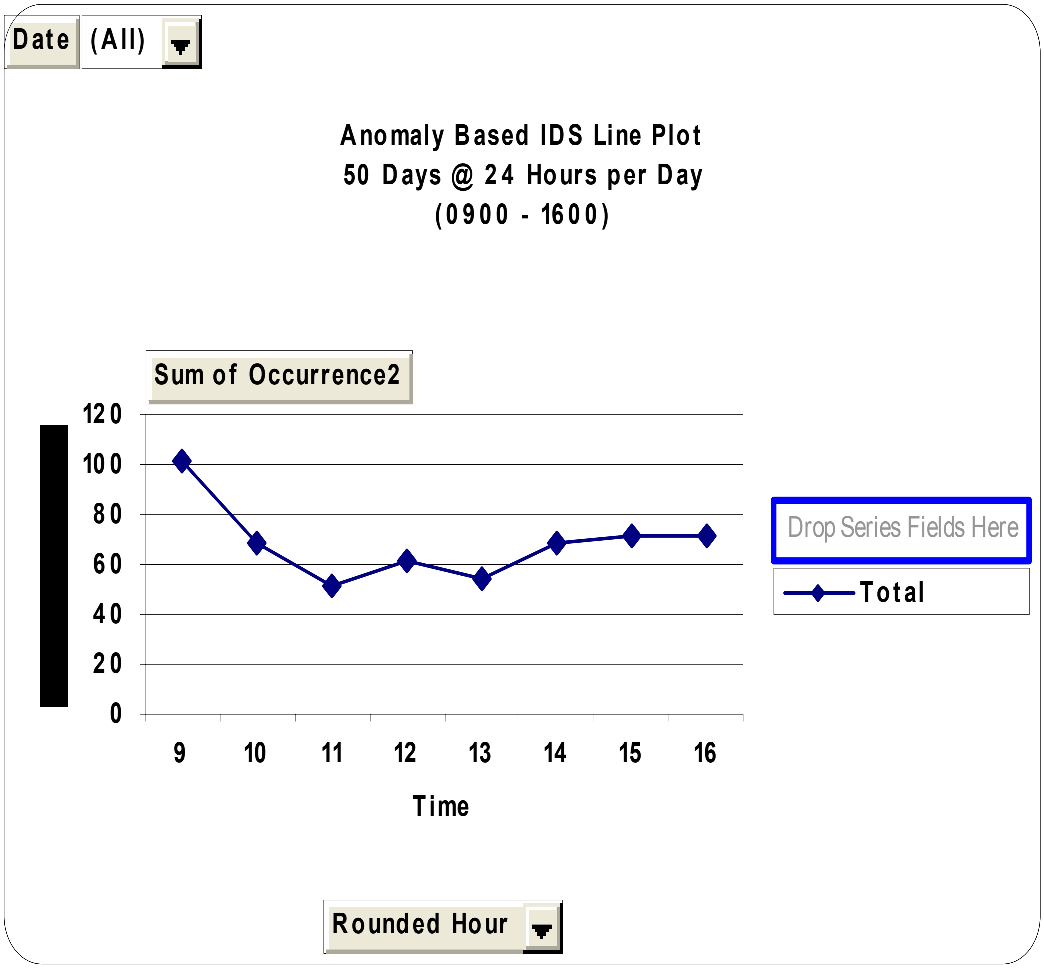


Figure 32. Anomaly Based IDS 0900 – 1600 Line Plot

3. 1700 – 2400 Time Frame Analysis

Table 14, below, represents this time frame with the number of mischievous occurrences per rounded hour. Figure 33 represents this.

1700 - 2400 Time Period		Standard Deviation
Rounded Hour	Total	17.62
17	78	
18	60	Mean
19	76	59.75
20	72	
21	53	Median
22	34	65
23	70	
24	35	Mode
Grand Total	478	#N/A

Table 14. Anomaly Based IDS 1700 – 2400 Time Frame

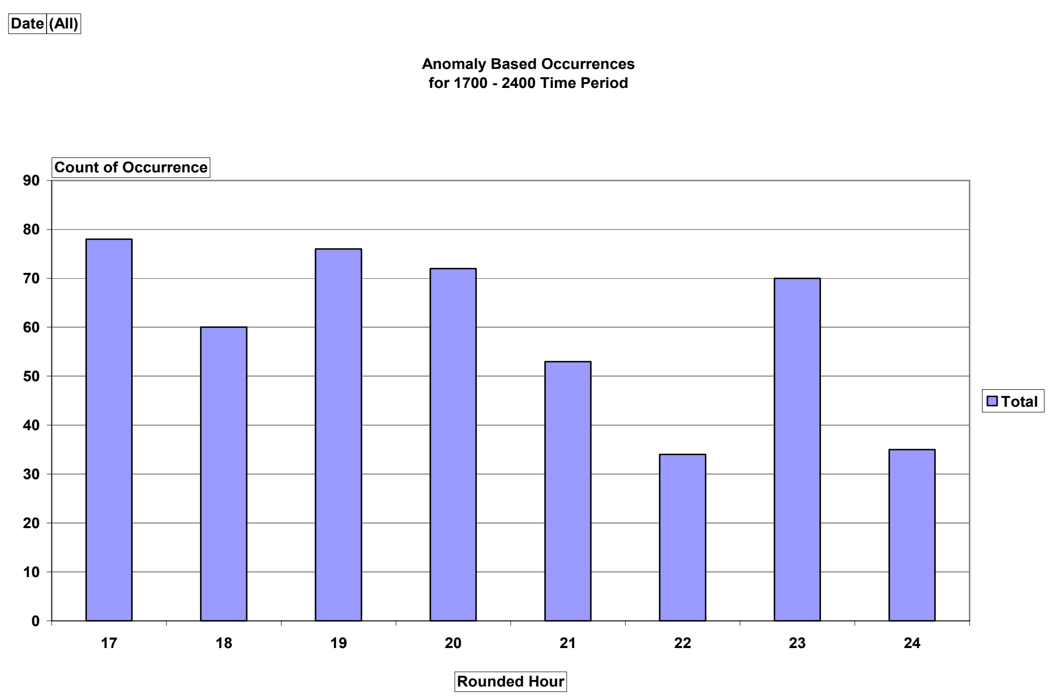


Figure 33. Anomaly Based IDS 1700 – 2400 Bar Graph

A plot of these occurrences per rounded hour reveals that, for this time period, only 62.5% of occurrences fall within σ of μ.

Figure 34 represents Table 14's statistical data.

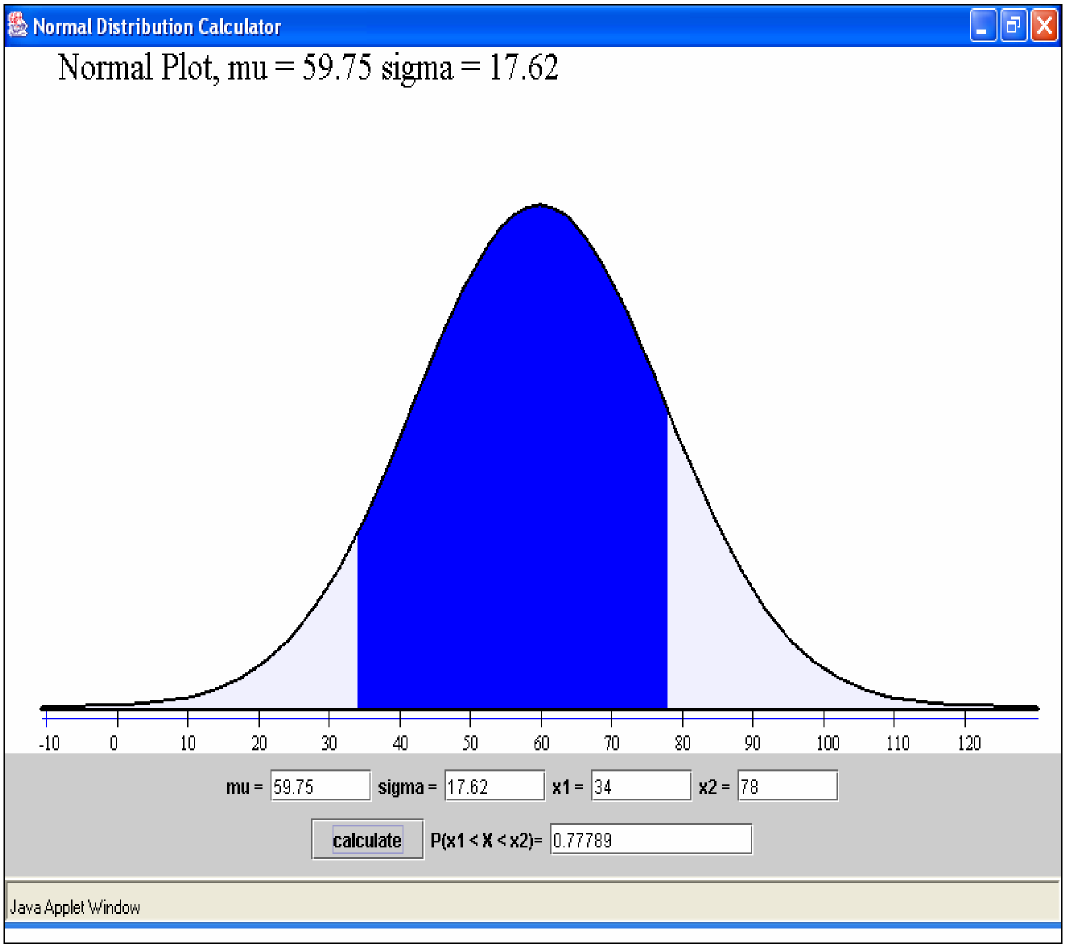


Figure 34. Anomaly Based IDS 1700 – 2400 Distribution Curve [From: CSUSB-04]

Figure 35, below, represents a line plot of these mischievous occurrences collected during the 50 day test date range specifically for this time period.

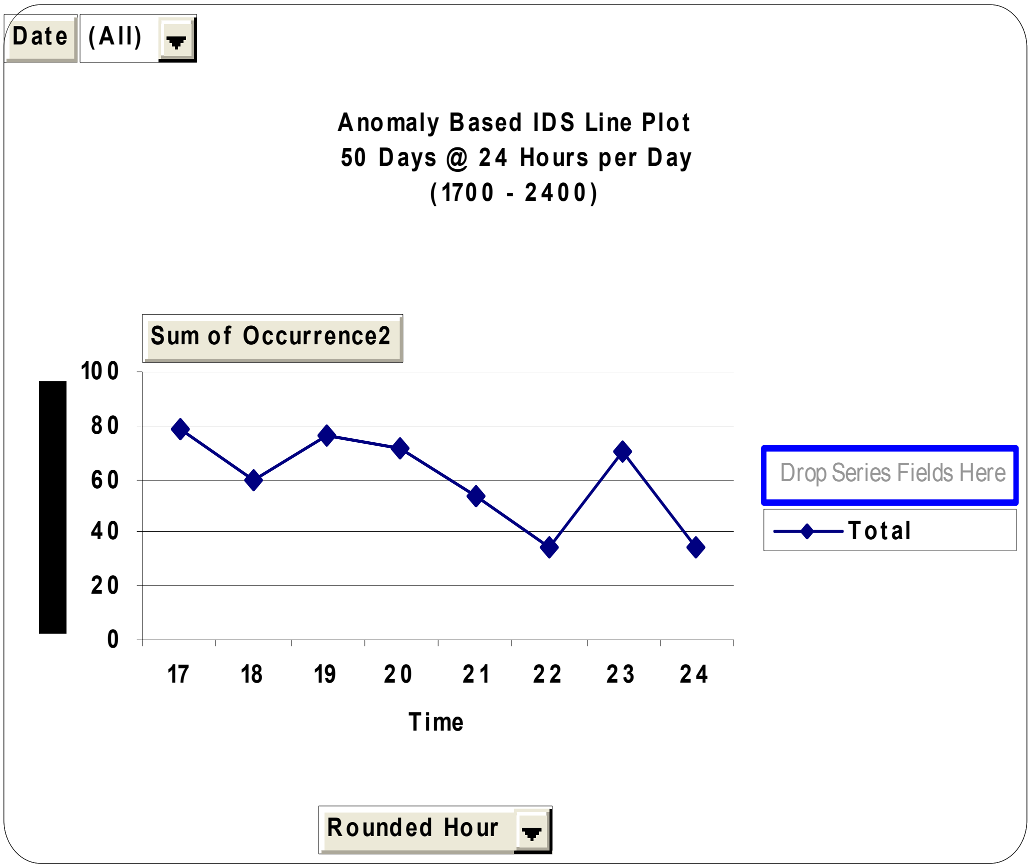


Figure 35. Anomaly Based IDS 1700 – 2400 Line Plot

4. Anomaly Based IDS Cumulative 24 Hour Period

Table 15 below, represents this cumulative period with the number of mischievous occurrences per rounded hour. Figure 36 represents this in a bar graph.

24 Hour Time Period		Standard Deviation
Rounded Hour	Total	18.45
1	63	
2	42	Mean
3	30	57.04
4	26	
5	33	Median
6	41	57.50
7	52	
8	54	Mode
9	102	72
10	69	

11	51	Range
12	62	76
13	55	
14	68	Minimum
15	71	26
16	72	
17	78	Maximum
18	60	102
19	76	
20	72	Standard Error
21	53	3.766
22	34	
23	70	Confidence Level (95.0%)
24	35	7.790
Grand Total	1369	

Table 15. Anomaly Based IDS Cumulative 24 Hour Period

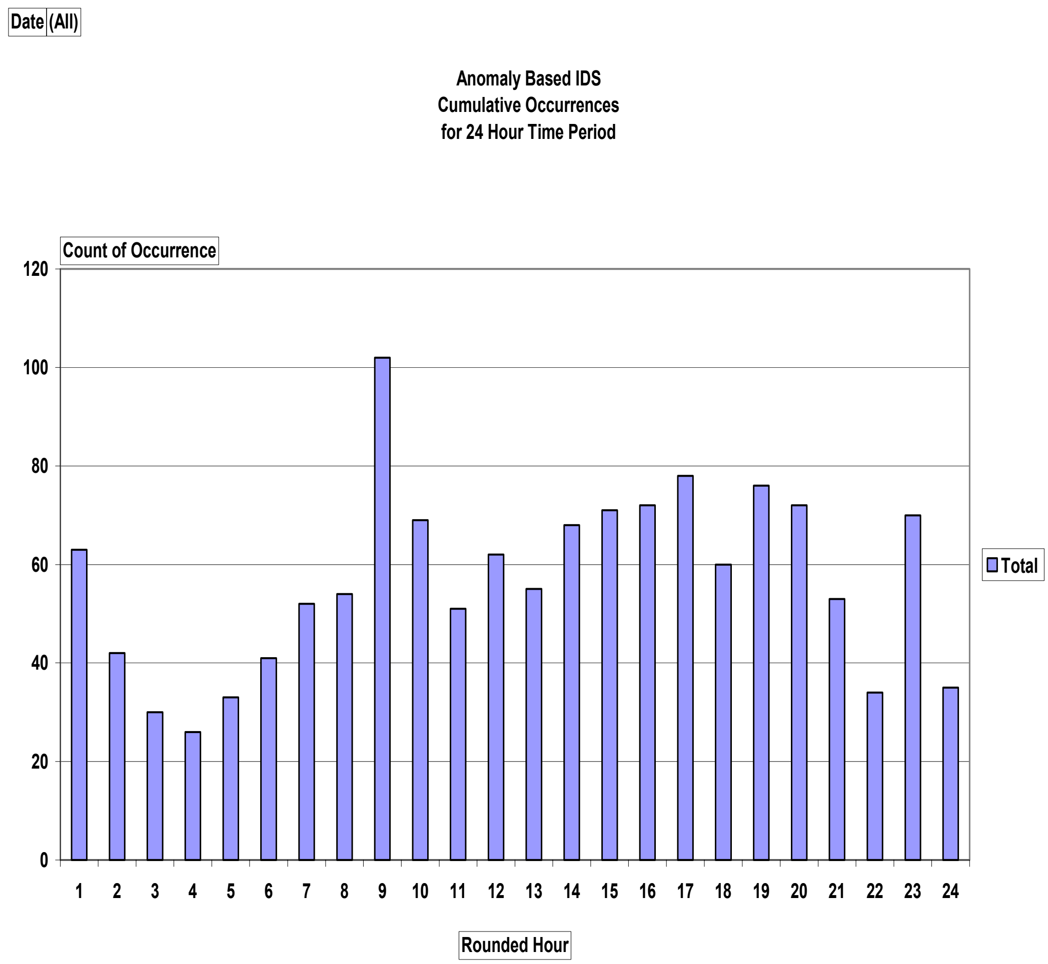


Figure 36. Anomaly Based IDS Cumulative Bar Graph

A plot of the mischievous occurrences per cumulative time frame reveals that, for this time period, 66.6% of occurrences fall within σ of μ. This low percentage makes it easy to assume the Null Hypothesis will be found statistically significant.

Figure 37, below, represents Table 15's statistical data.

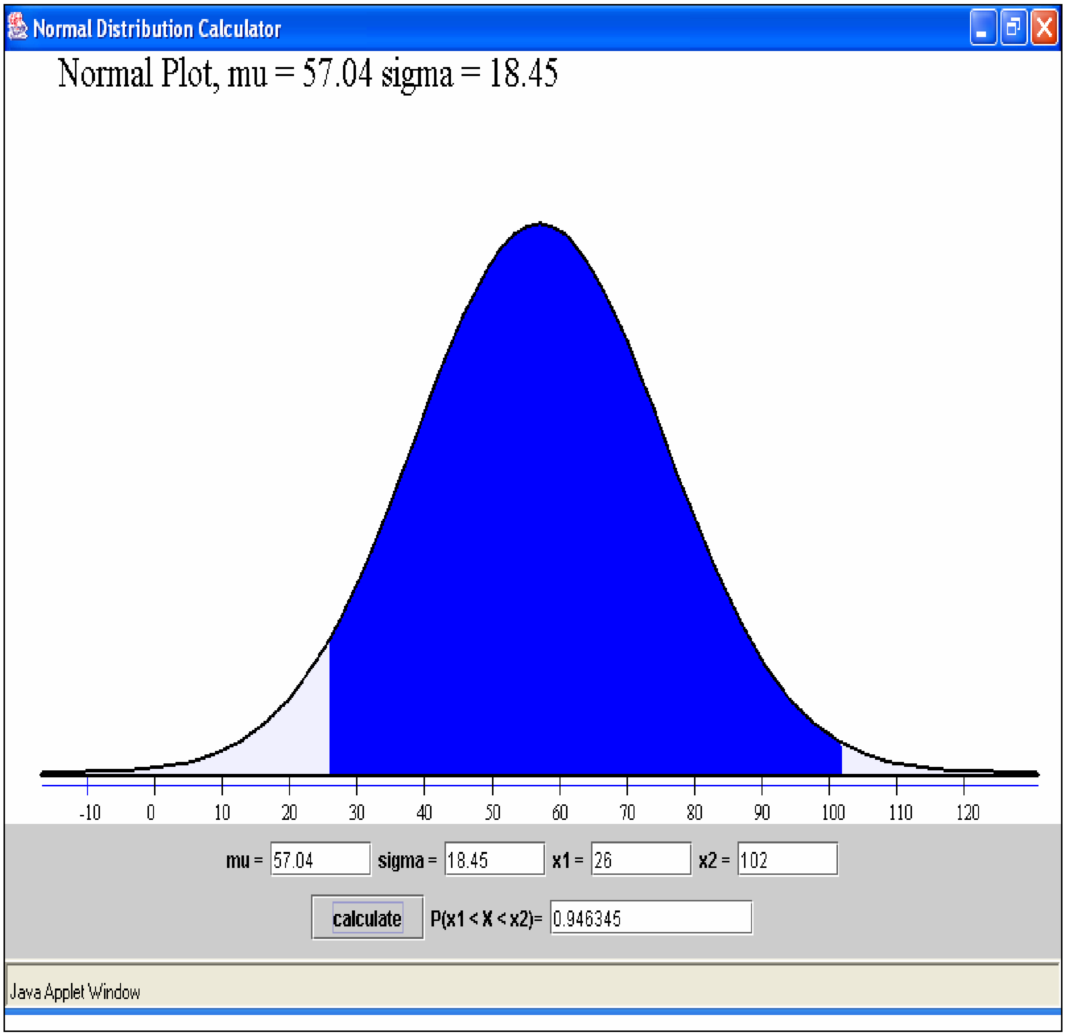


Figure 37. Anomaly Based IDS Cumulative Distribution Curve [From: CSUSB-04]

Figure 38, below, represents a line plot of these mischievous occurrences collected during the 50 day test date range specifically for this time period.

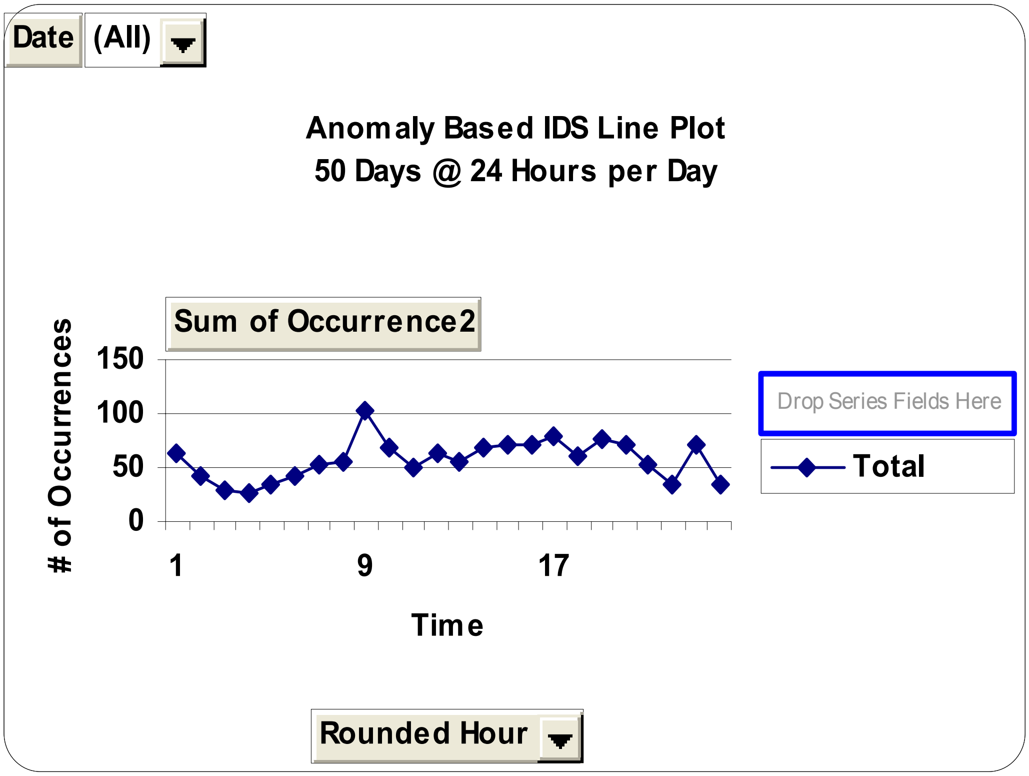


Figure 38. Anomaly Based IDS Cumulative 24 Hour Period Line Plot

The preceding figure reveals that, as previously stated, most roguish occurrences were *recorded* in the 0900 – 1600 time period.

E. SIGNATURE BASED/ANOMALY BASED IDS CUMULATIVE COMPARISON

The preceding two sections introduced the reader to nefarious traffic, that when analyzed, affords one the ability to make statistical inferences to the performance levels of both IDS's. This section will cross-analyze that data.

1. Combined Line Fit Plot

Figure 39 is a combined line plot that shows a graphical representation of the wayward traffic collected for both systems over the entire test date range. Although similar in appearance, there are a few several differences.

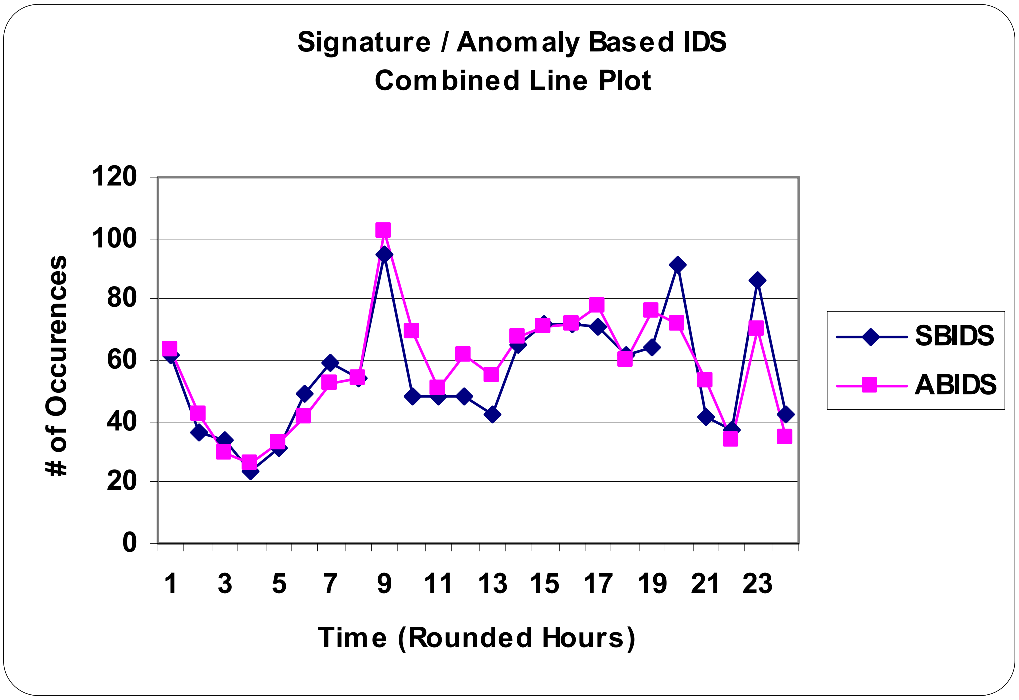


Figure 39. Anomaly Based/Signature Based IDS Combined Line Plot

As one can see, the diamond-line represents the signature based IDS while the square-line represents the anomaly based IDS. The sporadic peaks noted around hours 9, 20, and 23 on the blue line suggests increased activity, however, it is presumed these types of IDS's will have this pattern since they work off known signatures. Since a signature is easy to recognize, it does not take much nefarious activity to trigger the alarm. A good analogy would be a "hair" trigger on a revolver.

The pattern presented by the pink line suggests that since the anomaly based system must first develop a baseline of the network, it runs like a well-paced engine and easily captures traffic that steps outside this baseline. It then returns to status quo once the activity has passed.

2. Regression Analysis

Before delving into the statistics generated from the experiments regression analysis test, a Least-Squares Regression Line and R^2 value interpretation refresher is in order.

The Least-Squares Regression Line is the line that minimizes the sum of the squares of the vertical distances of the observed *y*-values from the line to as small as possible. If the variables are perfectly correlated (r =1 or r = -1), then the change in the predicted response $\hat{y}$ (pronounced y-hat) is the same (in standard deviation units) as the change in *x*. If $-1 \leq r \leq 1$, the change in $\hat{y}$ is less than the change in *x* [MOORE-01]. This means if the data points lie closer to the line, there is better fit and data correlation.

R^2 (square of the correlation) is explained as the fraction of the variation in the values of *y* that is explained by the Least-Squares Regression of *y* on *x*. This square basically gives a better feel for the strength of the association. Perfect correlation (r = -1 or r = 1) means the points lie exactly on the line. If $R^2 = 1$, then the variation in one variable is accounted for by the linear relationship with the other variable. If r = -0.7 or r = 0.7, then $R^2 = .49$ and about half the variation is accounted for by the linear relationship. Using the .49 example shows that .7 correlation is about halfway between 0 and $\pm$ 1 [MOORE-01].

The signature based IDS line fit plot found in Figure 40 suggests that with an increase in time of day there is tendency to move farther from the mean. By analyzing the R^2 value (.1283) found in Table 16 and Figure 40 helps explain there is about a 13% link between time of day and number of occurrence. Therefore, although the regression line shows some correlation, the low R^2 value demands another test to find a favorable outcome.

Signature Based IDS Summary Output	
Regression Statistics	
Multiple R	**0.36**
R Square	**0.13**
Adjusted R Square	**0.09**
Standard Error	**18.20**
Observations	**24.00**

Table 16. Signature Based IDS Summary Output

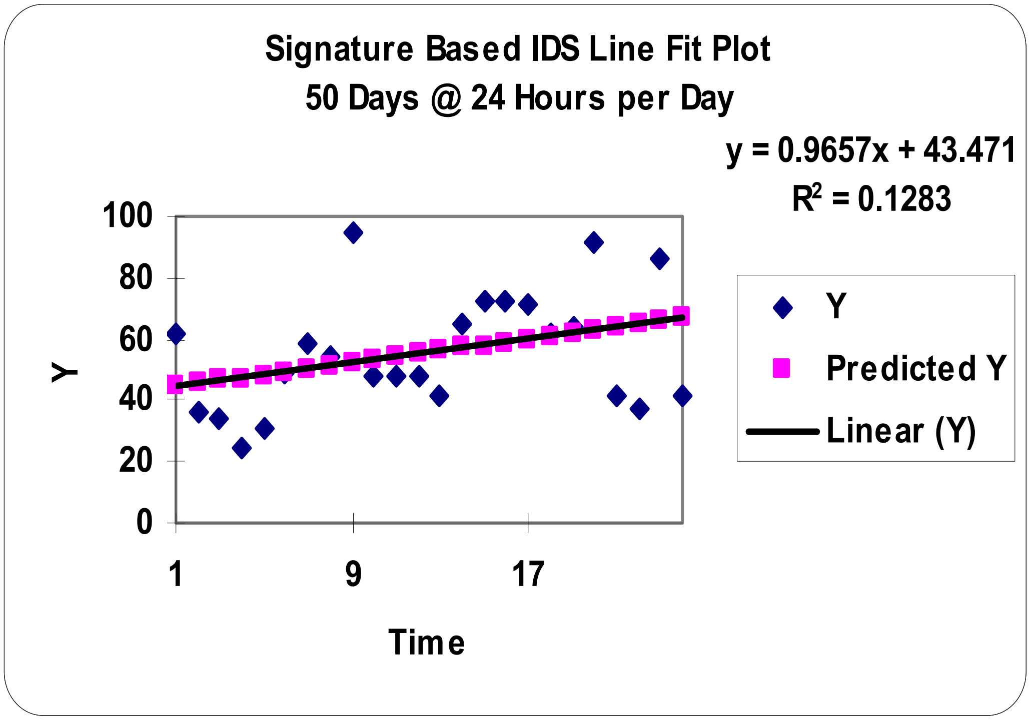


Figure 40. Signature Based IDS Line Fit Plot

Conversely, the anomaly based IDS line fit plot found in Figure 41 suggests that with an increase in time of day there is tendency to move closer to the mean. However, analyzing the R^2 value (.0772) found in Table 17 and Figure 41 explains there is about an 8% link between time of day and number of occurrence. Therefore, although the regression line shows much stronger correlation than the signature based IDS regression line, the low R^2 value also demands another test.

Anomaly Based IDS Summary Output	
Regression Statistics	
Multiple R	**0.28**
R Square	**0.08**
Adjusted R Square	**0.04**
Standard Error	**18.12**
Observations	**24.00**

Table 17. Anomaly Based IDS Summary Output

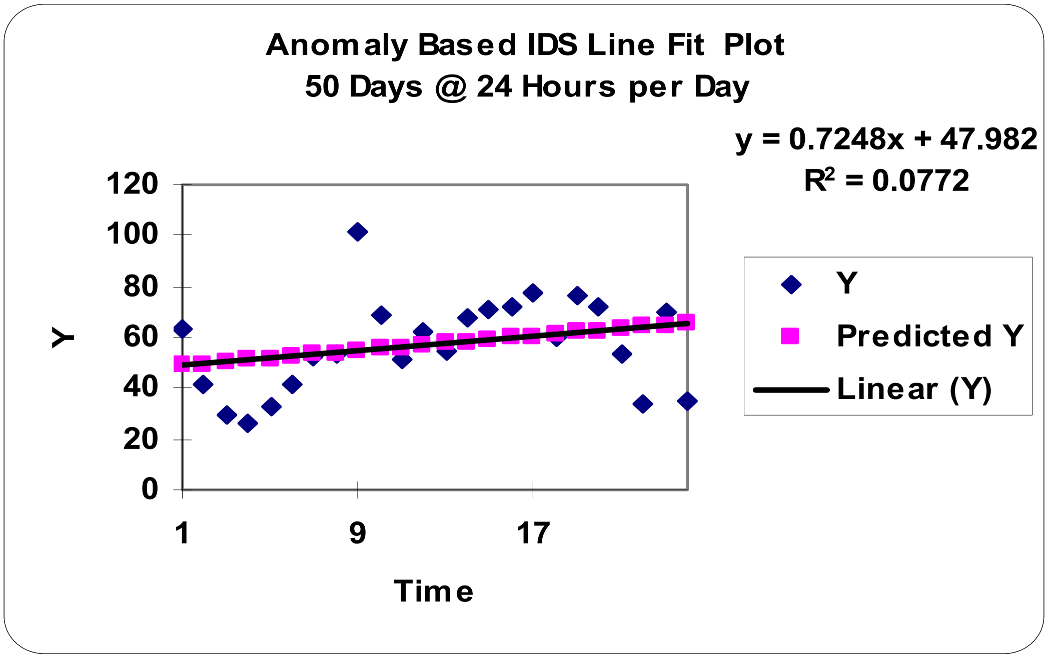


Figure 41. Anomaly Based IDS Line Fit Plot

3. Test of Statistical Significance

A test of significance finds the probability of getting an outcome *as extreme or more extreme than the actually observed outcome* [MOORE-01].

Originally stated in the hypothesis, a significance level (α) of .05 will be used as the measure to prove the Null Hypothesis statistically insignificant. With this level, we are requiring that the data give evidence against Ho (Null Hypothesis) so strong that it happens no more than 5% of the time when Ho is true. Therefore, in order to prove the Null Hypothesis as statistically insignificant the P-value (one-tail) that results from a significance test must be $\leq$ to α.

It was determined that the z-Test is best fit for this comparison. The z-Test makes the following assumptions: A Sample Random Size (SRS) of *n*, known population standard deviation σ, and either a normal population or a large sample. The SRS used for this test was the cumulative data tables of both IDS's.

Table 18 shows the results of the z-Test.

z-Test: Two Sample for Means		
	Variable 1	*Variable 2*
Mean	**57.04**	**55.54**
Known Variance	**340.30**	**363.48**
Observations	**24.00**	**24.00**
Hypothesized Mean Difference	**11.00**	
z	**-1.75**	
P(Z<=z) one-tail	**0.04**	
z Critical one-tail	**1.64**	
P(Z<=z) two-tail	**0.08**	
z Critical two-tail	**1.96**	

Table 18. z-Test of Significance

F. CHAPTER SUMMARY

This chapter discussed the data normalization process, which included the collection period, the process that determined data legitimacy, and how the legitimate data was normalized into 8-hour time blocks. It analyzed the mischievous traffic collated by both the signature based and anomaly based IDS's and represented this data in statistical tables, normal distribution curves, and line plots. It concluded with a cumulative comparison of both IDS's which included a line plot analysis, regression analysis, and a z-Test of significance.

V. EXPERIMENT ANALYSIS AND CONCLUSION

In the field of observation, chance favors only the mind that is prepared.

Louis Pasteur

A. PROLOGUE

This chapter will articulate the results found in Chapter IV, Section E, signature based/anomaly based IDS cumulative comparison. It will first examine and draw inferences regarding the line fit plots of both IDS's. This will be followed by a synopsis of the regression analysis and the results found in the z-Test of Significance. The chapter will then make remarks regarding the Null or Alternative Hypothesis as being statistically significant. It will conclude with a general thesis summary.

B. EXPERIMENT ANALYSIS AND CONCLUSION

1. Synopsis of Line Fit Plots

The information generated from the line fit plots affords the reader the ability to formalize an opinion regarding each IDS's nefarious traffic. As previously mentioned, the time period with the most traffic occurrences for the signature based IDS was 1700 – 2400 while the busiest time period for the anomaly based IDS was the 0900 – 1600 time block. The introduction of the graphical line fit plot helps the reader visualize this activity and also helps to surmise a conclusion. Furthermore, when overlaid, the line fit plots show both similarities and differences of traffic captured by both IDS's. However, this does not provide conclusive evidence to support either hypothesis. It was used to provide a visual reference for the reader and further testing is warranted.

2. Synopsis of Regression Analysis

When a table is presented that portrays regression data, it is not easy to visualize the emerging data pattern. However, when one is able to see a regression line along with its associated plot points, one can easily see the overall trend of this data. A regression line helps to fit the data by minimizing the sum of the squares of the vertical distances of the observed y-values from the line. If the points are close to the regression line, then there is evidence supporting a tight fit and strong correlation. The regression lines developed for both IDS's clearly indicate a favorable tendency towards the anomaly based IDS and the alternative hypothesis. However, this is still not enough conclusive evidence to support this inference.

In addition, it was previously explained that the closer the R^2 (square of the correlation) is to $\pm$ 1, the more strength of association, and all of the variation in one variable (time) being accounted for by the linear relationship with the other variable (# of occurrences). The R^2 values for the signature based and anomaly based IDS are .1283 and .0772, respectively. Since these values are close to zero, this method also can't be used to solely provide a conclusion and therefore, another test must be used.

3. Synopsis of the z-Test Analysis

The z-Test of significance was the last test used to find support of either hypothesis. As previously mentioned, the z-Test assumes that a Sample Random Size (SRS) of size *n*, with a known population standard deviation σ, and either a normal population or a large sample will be used. These assumptions fit the cumulative data tables of both IDS's.

Statistical inference contends that in order to prove the Null Hypothesis statistically insignificant, a p-value must be generated that is $\leq$ the significance level (α) used. This thesis used .05 for α and upon conclusion of the z-Test, the p-value generated was .04. This fact in itself proves the Null Hypothesis as statistically insignificant and shows favor in support of the Alternative Hypothesis.

C. CHAPTER SUMMARY

This chapter reiterated the information generated from the line fit plots and regression analysis and furthermore discussed each IDS's test results, formalizing a conclusion. It then re-visited the z-Test that provided conclusive evidence in support of the Alternative Hypothesis. It made reference to the p-value (.04) that was calculated which provided the necessary substance to discount the Null Hypothesis as statistically insignificant.

Furthermore, if this p-value is paired with the line-fit plots and the information generated from the regression analysis, there is apparent and overwhelming evidence that anomaly based IDS's not only are required in a Defense-in-Depth environment, but also provide substantial return-on-investment. This also provides conclusive evidence to rule out Type I and Type II errors.

D. THESIS SUMMARY

As one can tell, the anomaly based IDS kept pace and at times, simply out-worked the signature based IDS. With new or yet unknown viruses (aka: zero- day attacks), the signature based system is left idle until provided a virus signature or pattern. Whereas, the anomaly based IDS, during this same zero-day attack, has already sounded the alarm and notified the security manager or system administrator. Additionally, a point must be made that the turn-around time from the initial zero-day occurrence to the time when a vendor supplies an identifying pattern signature has been significantly reduced. What used to take weeks is now only a matter of days.

It was not this thesis' intent to review a particular name-brand of IDS. This thesis focused more on the holistic view or approach to intrusion detection and how those types of IDS's work. There was no favoritism or bias towards the two systems used for testing. The author would have gladly used other available means. Furthermore, this thesis was conducted with no financial interest or gain from any vendor.

THIS PAGE INTENTIONALLY LEFT BLANK

VI. FUTURE WORK AND RECOMMENDATIONS

One small step for man, one giant leap for mankind.

Neil Armstrong, 1969

A. PROLOGUE

This chapter will present areas the author feels warrants future study.

B. RECOMMENDATIONS

1. Security Switches

These devices are synonymous with intrusion prevention devices, security blades, or other specialty appliances. Whatever their name, it is clear there is an evolution towards dedicated hardware that optimizes security functionality into a single box, blade, or chip [INFOSEC-03]. For further reading, follow this hyperlink to the Information Security, November 2003 article, p. 59.

http://infosecuritymag.techtarget.com/ss/0,295796,sid6_iss205_art412,00.html

2. Target Based Intrusion Detection Systems

These systems squelch network noise to pinpoint the alerts you care about. Target-based IDS is a new technology that correlates knowledge about network topology, operating systems, and applications with incoming attack information [INFOSEC-04]. For further reading, follow this link to the Information Security, January 2004 article, p. 35.

http://infosecuritymag.techtarget.com/ss/0,295796,sid6_iss306_art540,00.html

3. Intrusion Prevention Systems

As discussed in Chapter II, IPS's are considered the next logical step in the evolution of IDS's. These systems are the combination of the blocking capabilities of firewalls with the deep packet inspection capability of IDS's. An IPS is defined as "any device (hardware or software) that has the ability to detect attacks, both known and unknown, and prevent the attack from being successful" [DESAI-03]. For further reading, follow this link to a study conducted by the Network World Fusion Magazine.

http://www.nwfusion.com/reviews/2004/0216ips.html

4. Protocol Anomaly Detection

A new variant of anomaly detection has been incorporated into IDS's in recent years. Instead of training models on normal behavior, protocol anomaly detectors build models of TCP/IP protocols using their specifications. Statistical anomaly detection is plagued by the inability to create a normal model of network traffic statistics. Protocol anomaly detection is much easier, however, because protocols are well defined and a normal "use" model can be created with greater accuracy. Protocols are created with specifications, known as RFCs, to dictate proper use and communication. All connection oriented protocols have state. Certain events must take place at certain times. As a result, many protocol anomaly detectors are built as state machines. Each state corresponds to a part of the connection, such as a server waiting for a response from a client. The transitions between the states describe the legal and expected changes between states [DAS-01]. For further reading, follow the link to the SANS article.

http://www.sans.org/rr/papers/30/349.pdf

5. Collaborative Intrusion Detection Systems

CIDS employs multiple specialized detectors at the different layers - network, kernel and application - and a manager based framework for aggregating the alarms from the different detectors to provide a combined alarm for an intrusion. For further reading, follow the link to the IEEE article.

http://csdl.computer.org/comp/proceedings/acsac/2003/2041/00/20410234abs.htm

C. CHAPTER SUMMARY

This chapter introduced the reader to areas that might be of interest or useful in complementing those devices used in a Defense-in-Depth environment.

APPENDIX. COMPLEMENTARY INTRUSION DETECTION SYSTEM EXPERIMENT

A. PROLOGUE

The following article from InfoWorld Magazine is an IDS comparison/experiment the author was involved with. Several IDS's, to include signature, anomaly based, and hybrids were provided by vendors for analysis. The comparison ran parallel to the author's IDS experiment and therefore provided valuable insight for his experiment. Although different metrics were used to analyze the IDS's, the knowledge gained from establishing the network, installing and configuring all systems, and interacting with the vendors proved invaluable.

B. INFOWORLD ARTICLE


INSPECTING
STEALTH WATCH
Lancope
peakflow X
ARBOR
StillSecure
INTERNET SECURITY SYSTEMS
proventia
38 INFOWORLD.COM 08.23.04

Network Detectives

THE INSPECTORS

ISS, Lancope, Snort, and StillSecure detection systems sniff out network threats

JUST A FEW SHORT YEARS AGO, AN IDS WAS A LUXURY. BEFORE THE RISE OF THE WEB APPLIcation and the worm, most networks were adequately defended by a firewall at the perimeter and a virus scanner at the mail server. Today, the firewall remains effective against clumsy DoS attacks and run-of-the-mill exploits, but it's hard-pressed to thwart application-layer attacks that piggyback on welcome protocols and worms that wind their way inside the network through any overlooked port or a mobile user's laptop.

Not only are perimeter defenses less adequate than they used to be, but internal network resources — including business-critical applications exposed to the Web — are more valuable to their companies than ever. Naturally, the double whammy of a hole-ridden perimeter and an invaluable core has network managers looking for an edge. The IDS is becoming part of the standard toolkit.

We tested four network IDS products in May, June, and July at the Naval Postgraduate School in Monterey, Calif., pitting Internet Security Systems (ISS) Proventia G200, Lancope StealthWatch 4.0, Snort 2.10, and StillSecure Border Guard 4.3 against both live Internet traffic and a variety of attacks we launched from penetration testing tool Core Impact 4.0.

Our manual attacks included OS fingerprinting, privilege escalation, DoS, banner grabbing, traversal attacks, and Microsoft IIS and Apache Web server exploits, among others. More significantly, on the live network, the products were exposed to nearly a thousand unique "attackers" targeting more than 50 ports, detecting thousands of "events" coming in from the Internet or from several thousand hosts inside the network. Among the live threats our IDS products confronted were the Sasser worm and Gator spyware.

As we expected, all four products did a good job detecting threats. With only one exception, in which one IDS initially failed to identify the Sasser worm, the products successfully alerted us to the presence of all the manual attacks and live threats they confronted. Although the four proved roughly equal in terms of recognizing attacks, important differences — ranging from ease of setup and management to depth of packet analysis and reporting, but especially the fundamental approach taken in detect-

BY VICTOR R. GARZA AND JOSEPH L. ROTH | PHOTOGRAPH BY KEVIN CANDLAND ing threats — may help dictate which solution best suits your network.

Snort With ACID

Snort is the famous free and open source IDS. It's supported by an active community of users and developers who regularly and promptly update Snort's signatures in response to newly discovered threats. Snort is a great choice if you have more time than money. When regularly maintained, Snort can be very effective. The downside is that maintenance doesn't come easy. Snort requires care from a dedicated expert, and you'll need to roll up your sleeves and wrestle with a difficult installation and setup.

You can pull all the files you need off the Snort project (snort.org), where you'll also find many tutorials, FAQs, and Snort manuals to help you out. The standard installation of Snort — ACID (Analysis Console for Intrusion Databases); PHP, which is required by ACID; and MySQL on Red Hat Linux — is the best-documented. A Windows XP installation is also well-documented. Deviations such as Windows 2000 and Microsoft SQL Server 2000 aren't supported as thoroughly.

There are three run modes for Snort: Sniffer, Packet Logger, or NIDS (Network IDS). It's easy to operate in any mode. We installed Snort on both Windows XP and Red Hat Linux 9.0, running both instances in NIDS mode. The Windows XP installation requires installing WinPcap 3.0, an architecture for packet capture and network analysis, before installing Snort. We also installed Barnyard, a free plug-in that offloads Snort logging, helping to accelerate Snort's packet processing and thereby alleviate packet loss.

Snort's strength is its high degree of configurability. Its main weakness is its dependence on (sometimes poor) signatures. As with all signature-based IDSes, Snort can be defenseless against unknown or "zero-day" attacks until a signature becomes available. Another problem with Snort is that some of the signatures — no doubt designed to identify older attacks — look for benign words (such as "TOP") in the payload to determine whether a packet is malicious. As a result, an initial ruleset from the Snort project gave us several hundred false positives.

Snort developers have addressed this drawback by allowing you to comment out rules that you do not want to use on

Border Guard 4.3
StillSecure stillsecure.com

EXCELLENT	8.6
Threat Detection (30%)	8
Management (20%)	8
Ease of use (10%)	9
Scalability (10%)	9
Security (10%)	10
Setup (10%)	9
Value (10%)	9

COST: Starts at $7,500 for device and $1,500 per year for maintenance (subscription option available)

PLATFORMS: Management console: Windows, Internet Explorer 6 or later

BOTTOM LINE: Border Guard brings ease-of-use, multinode management, and intrusion prevention capabilities to Snort. Installation and setup are fast and easy, the GUI is top-notch, and reporting is excellent, removing all the difficulty of navigating Snort and displaying attacks and payloads. An excellent choice for signature-based detection and prevention.

Proventia G200
Internet Security Systems iss.net

VERY GOOD	7.8
Threat Detection (30%)	8
Management (20%)	7
Ease of use (10%)	8
Scalability (10%)	9
Security (10%)	10
Setup (10%)	6
Value (10%)	7

COST: Starts at $11,995

PLATFORMS: SiteProtector management console: Windows 2000, Windows XP, Windows Server 2003

BOTTOM LINE: Proventia combines signature-based detection and prevention capabilities with a depth of packet analysis unmatched by its competitors, making it a good solution for monitoring and enforcing network policies. Time-consuming configuration and a complex management interface, however, make Proventia less suitable as an everyday IDS.

Snort 2.10 with ACID
Snort.org snort.org

VERY GOOD	7.3
Threat Detection (30%)	7
Management (20%)	6
Ease of use (10%)	7
Scalability (10%)	8
Security (10%)	9
Setup (10%)	6
Value (10%)	10

COST: Free

PLATFORMS: Linux, 32-bit Windows, BSD, Mac OS X

BOTTOM LINE: Snort is a free, flexible, effective rules-based IDS that is difficult to set up and not particularly user-friendly. Multisystem management isn't supported, and reporting and management fall short of commercial offerings. On the plus side, you can use existing rules, which are regularly updated by an active open source community, or configure your own.

StealthWatch 4.0
Lancope lancope.com

EXCELLENT	8.9
Threat Detection (30%)	9
Management (20%)	9
Ease of use (10%)	9
Scalability (10%)	9
Security (10%)	10
Setup (10%)	8
Value (10%)	8

COST: Starts at $9,995 for M45 appliance

PLATFORMS: Web management interface: Internet Explorer 6.0 or later, Netscape 6.2 or later

BOTTOM LINE: StealthWatch tunes into deviations in normal network traffic and host behavior, an approach that enabled it to warn of a Sasser worm outbreak on the test network ahead of our signature-based detection systems. On the downside, networking expertise is required to use StealthWatch effectively; novice administrators will be challenged.

Despite limitations, Snort has a loyal following — and for good reason.

your network. The problem with this is, anytime you update your rules with the newest set from Snort.org, you'll have to comment them out again. Oinkmaster, an open source Perl script, automates the process of enabling and disabling specified rules after each update. It was designed to run easily on Unix or Linux, but using it in a 32-bit Windows environment requires that ActivePerl, GNU, and GNUwget be installed.

We liked the fact that we could use the detection rules that came with Snort or roll out our own. Snort logs packets that are flagged by Snort rules. The rules themselves are configured in a hierarchical structure and do a good job of capturing suspicious traffic. When Snort logs in binary mode, it logs the packets in tcpdump format to a single file in a designated directory. This is especially useful in large installations that will include additional analysis with the Ethereal protocol analyzer, for example.

ACID is a graphical front end for Snort. Using it isn't strictly necessary, and it was painful to install on Windows XP and IIS 5.0 because it also required the installation and configuration of PHP and the JpGraph graph library for PHP. But ACID is a powerful tool for handling Snort alerts, and it makes a good alternative to analyzing raw Snort data from the command line. ACID can query Snort's binary log files or a MySQL, PostgreSQL, Oracle, or Microsoft SQL Server database.

The reporting offered by Snort and ACID was better than we expected. This was especially true when it came to ACID's graphical reporting, which can chart information based on date, signature, protocol, IP address, port, and so on. We liked how, at the end of each user session, ACID presented an informative graph of traffic statistics.

A free IDS offers a lot of flexibility. We didn't have to think twice about creating IDS redundancy on our test network by having distributed Snort boxes monitoring different subnets. We also liked that we could specify a particular machine on our network for log storage. The downside is that there's no way to centrally control multiple Snort consoles.

Snort doesn't use the NMAP (Network Mapper) port scanner to map the network but instead relies on packet sniffing, so there's no risk of locking up or crashing a host. But packet sniffing also doesn't provide as much detail as active fingerprinting.

Snort will require hours of configuration to tune out false positives, and its rules must be managed carefully. But it has a loyal following for good reason. Every large network should be running some kind of rules-based IDS, and Snort gets the job done.

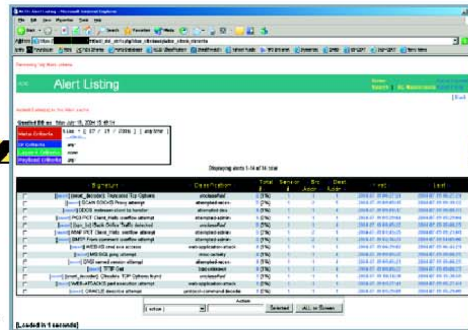
StillSecure Border Guard

StillSecure's Border Guard is a commercial product built on Snort. It offers an enhanced form of signature-based protection without the painful, time-consuming installation process, endless front-end configuration, and arduous

Network IDS Dossier Key differentiators between competitors include the method of detection used and whether the solution offers firewalling capability. Both Border Guard and Proventia can be deployed in-line to actively prevent attacks.

	Platform	Management interface	Multiconsole management	Detection method	Custom rules	Attack blocking	Packet logging	Application-level filtering	Alarms
Border Guard 4.3	Linux appliance or software	Web	Yes	Signatures	Yes	Yes	Yes	Yes	Yes
Proventia G200	Linux appliance	Windows client	Yes	Signatures	Limited	Yes	Yes	Partial	Yes
Snort 2.10 with ACID	Linux, Windows	Web	No	Signatures	Yes	No	Yes	Yes	Requires ACID and SWATCH
StealthWatch 4.0	Linux appliance	Web	Yes	Anomalies	N/A	No*	Yes	No	Yes

*Integrates with Check Point and Cisco firewalls and routers to block attacks.



ACID brings a Web GUI to Snort. The Alert Listing provides the basic facts regarding flagged threats, including time, source, and target.

rules upkeep. Unlike Snort, Border Guard can also serve as an intrusion prevention gateway, using the rudimentary Linux iptables firewall to provide several layers of traffic blocking. The downside, of course, is that it's not free.

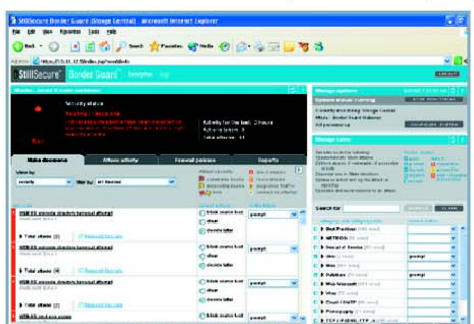
Sporting the best user interface of the four primary IDS products we tested, Border Guard has strong detection and reporting capabilities, including one interesting twist that it shares with Snort: the capability of sniffing out and reporting porn usage. This feature, which boils down to inspecting traffic for illicit keywords, can be especially helpful for identifying network utilization problems and enforcing company policy.

A StillSecure site engineer was present during our installation. We walked through the entire installation and configuration process and had the 1U appliance fully operational in less than 30 minutes. We also installed the Border Guard software on a PC at our satellite facility, turning a spare machine into a hardened appliance in 15 minutes.

We were immediately impressed by Border Guard's intuitive, easy-to-navigate tabbed interface. The main dashboard is tidy and understated. A stoplight in the upper left of the screen provides an at-a-glance view of overall security status. The Make Decisions tab lists the current attack or rule violation and offers options based on the severity level of the attack, including blocking the source host, clearing the alert, or deciding later. The Attack Activity tab shows a graph or table of total attacks and actions pending or taken.

Border Guard's reporting functionality and interface are excellent. Although exports are limited to only HTML, text, or CSV (comma-separated values) formats, we were impressed with the type and scope of reporting. Powerful filters make it easy to mine data in order to investigate specific attacks or offenders.

To ease initial setup, Border Guard provides a quick-tune option that is equivalent to a whitelist for instructing Border Guard to ignore threats to specific operating systems and hosts, such as Web or Microsoft Exchange servers, that are not on your network. Through quick-tuning, you can also configure Border Guard to ignore common traffic types, such as ICMP (Internet Control Message Protocol)



Border Guard provides a simple and straightforward dashboard for monitoring security status and responding to threats.

and SNMP, to reduce potential false positives.

Border Guard goes beyond Snort in other ways. It uses NMAP to actively identify nodes on the network, providing more accurate and detailed information. (A passive method of identifying hosts isn't provided.) It provides several layers of event notification, including e-mail alerts to identified recipients based on the severity of a detected attack or summary e-mails based on specified thresholds or attack limits. It stores backup settings in a Linux tar (tape archive) file, making configurations easy to recall and restore.

Border Guard also supports central, Web-based management of multiple nodes, where one node in the group becomes the master console. Using the multinode manager, a single ruleset can be configured and pushed out to all nodes or groups of nodes, a nice touch in a large environment.

Updating signatures is both flexible and granular. Options range from updating entire rulesets by automatically running a command script, to inserting a firewall policy, to logging or ignoring events. Border Guard allows rule updates to be installed automatically from the StillSecure database on an hourly basis, but we found every 12 hours to be more sensible.

Our Border Guard appliance crashed three times during testing. The first two crashes were caused by having filled up the appliance's hard drive, which was due to our setting the period of application payload capture to a lengthy five weeks. To fix the problem we had to call tech support to reindex the hard drive. Unless you have a large hard drive (200GB or bigger), we recommend using application payload capture sparingly or limiting the number of retention days to a week. Obvious factors, such as alarm settings and the makeup of your network traffic, will determine the appropriate capture settings for your enterprise.

According to StillSecure's tech support, the third crash was due to an incompatibility between the appliance's Dell hardware and the Border Guard software. The bug inadvertently causes the hard drive to become read-only, which prevents Border Guard from logging data and thus crashes the system. This only happened once during a month of testing but could be a significant problem. StillSecure acknowledged the bug and claims it will have a fix in the next version.

Thanks to an excellent interface, simple setup, and easy rules maintenance, Border Guard is well-suited to either the novice or the seasoned administrator. It offers all the benefits of Snort and more, without all the headaches.

ISS Proventia G200

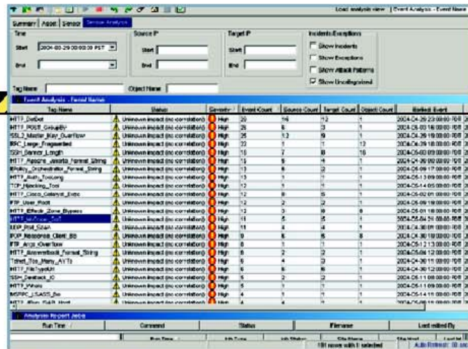
The Proventia G200 appliance from ISS can be deployed passively as an IDS or in-line as an IPS. Although the Proventia does a decent job of detection, we discovered that it seems better suited as a network analysis or auditing tool.

We found installation cumbersome due to Proventia's dependency on an external database for logging. We configured the Proventia as a passive network device, using a span port on our network to monitor all traffic flowing into and out of our test environment. In addition to IDS and IPS modes, the Proventia also offers an intermediate option called the "in-line simulation" mode. Here the sensor will just send alerts about things it would normally block in IPS mode, allowing you to test IPS policies before deployment.

In addition to setting up a separate Microsoft SQL Server database, which Proventia used as the primary repository for captured data, we had to install a Windows client — the SiteProtector console — on our management workstation in order to communicate with both the sensor and the database and to retrieve stored and correlated data. Unlike the other three competitors in this comparison, ISS does not provide a Web management interface.

We concede that the Windows client does enhance security because it creates a strict relationship between the appliance and the console. But it also restricts client platform options for administrators and limits the ability to distribute administrative duties, as each console requires the management client. We'd like to have the option of using a Web management interface.

SiteProtector was easy to configure, but it's completely dependent on the Proventia appliance and SQL Server database for all functioning and authentication. As we found out, if the SQL Server connection is not established, the appliance simply does not respond to console requests — not even with an error message. ISS should incorporate a pop-up message to inform the user when there is a problem.



ISS Proventia's SiteProtector console offers many options for analysis; it's easy to search for attacks by severity, frequency, and other criteria.

This wasn't the only usability hurdle we stumbled into. SiteProtector uses the database log-in and password combination established by the system administrator. If novice users attempt the log-in incorrectly, they are locked out without explanation.

On the plus side, the management console is designed to handle network vulnerability data from a variety of hardware and software sources, including ISS' host-based distributed client, RealSecure Desktop, and its vulnerability management software, Internet Scanner, which we reviewed last fall (infoworld.com/1752). The SiteProtector console also supports several other information-gathering tools, including the SiteProtector SecurityFusion correlation engine. SecurityFusion helps you prioritize defenses against possible attacks based on other ISS product data. We came to think of SiteProtector as a Swiss army knife of sorts.

During setup, Proventia presented us with lots of options for various network configurations. For example, we could create a policy that was geared for specific router traffic or traffic coming from a specific subnet. But we also found creating and managing policies to be slightly confusing and counterintuitive. Policies we created often didn't seem to justify the number of steps we were required to take — or the variety of templates we had to wade through — in order to get there.

Among all the products we reviewed,

we found that Proventia put the strongest emphasis on network traffic auditing, thanks to deeper protocol analysis capabilities than its competitors'. For example, if you had a zero-tolerance policy for FTP traffic, Proventia could easily supply you with the information necessary to combat violations, even going as far as capturing user names and passwords that are sent in the clear. Of course, the same auditing policy approach works with other types of traffic such as HTTP and POP3.

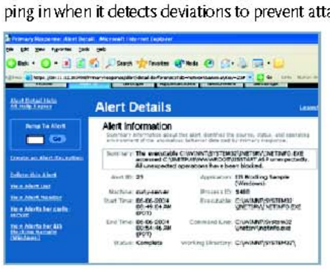
Application-layer traffic filtering in Proventia is extensive out of the box. The application auditing it provides is nonexistent in StealthWatch and would require creating custom rules in Border Guard and Snort. For example, we could filter on POP3 traffic and inspect headers for source and destination, and we could view quite a substantial amount of information regarding the session transaction — even when not viewing the actual payload.

Proventia also offers a plethora of options for reporting, including the ability to collect application-specific data such as successful FTP log-ons,

Attack of the Inhospitable Host

HOST-BASED INTRUSION DETECTION AND PREVENTION PRODucts are available from a smattering of vendors, including big guns such as Cisco, Internet Security Systems, McAfee, and Symantec, but Sana Security's Primary Response is the one that stands out, and for several reasons.

First and foremost, it is focused on protecting servers — more specifically, Microsoft Windows and Sun Solaris servers. In addition, Primary Response takes an innovative approach to application security, learning normal code paths taken during the execution of system calls, including local file access, and stepping in when it detects deviations to prevent attacks. And it can be installed and configured quickly and can be managed centrally via a Web browser.



Primary Response detects deviations from normal server behavior and sounds the alarm.

Primary Response consists of a management server and "adaptive profiling" agents. The agents run on your Windows or Solaris hosts, monitoring those servers and reporting back to the management box. We found that the product requires several days of "learning" before the agent can establish a baseline of normal application usage. Protection against buffer overflow attacks, however, is provided right out of the box without any need for tuning.

Primary Response is a breeze to manage. We liked the granular options for blocking file access during an anomalous event, and we appreciated the agent's ability to learn a server's behavior on an incremental basis and to "readapt" after an OS is patched, for example.

During our testing, while running Primary Response in learning mode, the product detected a breach of a Windows IIS server and the installation of a virus that caused a massive DoS attack on the local network. Sana's forensics tool helped us trace the attack to a system in Taiwan.

Primary Response provides effective host protection, but it would be nice if the product did more. For example, integration with a signature-based detection system would enable it to identify other potentially harmful occurrences rather than just those that are anomalous in nature.

It also struck us that, with an anomaly-based network IDS in place and the security features of Windows 2000 or Windows Server 2003 fully enabled, such host protection may not be necessary. But when a server is mission-critical, you don't take chances. For those who need airtight security, Primary Response provides a hedge against unknown vulnerabilities lurking in Windows and Solaris, as well as protection against insider attacks that a network IDS may not catch. — *V.R.G. and J.L.R.*

Mark A. Givens and Charles D. Herring of the Naval Postgraduate School contributed to this review.

Primary Response 2.2

Sana Security sanasecurity.com

VERY GOOD 7.9

Threat Detection (30%)	9
Management (20%)	6
Ease of use (10%)	9
Scalability (10%)	7
Security (10%)	10
Setup (10%)	7
Value (10%)	7

COST: $6,500 for the management server plus $1,750 per server agent

PLATFORM: Windows NT 4.0, Windows 2000, Windows Server 2003, Solaris 8

BOTTOM LINE: Primary Response blocks zero-day attacks, buffer overflows, and policy violations on Windows and Solaris servers. Agents are easy to install, learn normal host behavior automatically, and provide detailed information on attacks.

Proventia seemed better suited as a network analysis or auditing tool.

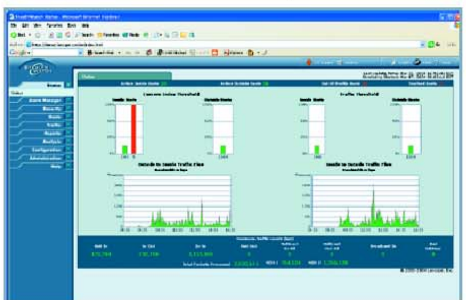
Telnet users and passwords, or HTTP session information. A 3-D pie chart of current traffic activity gives the user a quick overview and the ability to drill down into the details.

Proventia was the only IDS among the four that didn't catch the Sasser outbreak during testing. After we notified ISS, its engineers were able to trigger alerts off the Sasser signature, but even this took several attempts. Despite this shortcoming, Proventia earned higher marks in threat detection than Snort did, thanks to its avoidance of false positives. Proventia produced fewer false positives — but also fewer true positives — than either Snort or Border Guard.

Proventia is powerful and flexible but also complex. Its deep packet-analysis capabilities make it a good compliance-auditing tool, but the product didn't strike us as the best fit for straight intrusion detection. StillSecure's Border Guard is much better suited to that job, not only because it's easier to install and configure but also because it's more straightforward to maintain and monitor on an ongoing basis.

Lancope StealthWatch

Lancope's StealthWatch takes a different tack to detecting malicious activity than the other three IDS products we tested for this comparison. Instead of relying on signatures or predefined patterns to identify attacks, StealthWatch relies on anomalies — or exceptions to normal traffic trends — as indicators of a threat. This approach makes StealthWatch especially well suited to detecting worm outbreaks and exploits of unknown vulnerabilities.



The StealthWatch Status screen provides an at-a-glance, graphical overview of network traffic flows and an index of potential threats.

While all four of our IDS products were online for testing, StealthWatch alerted us to the potential Sasser outbreak before the other devices did. The downside to StealthWatch's approach is that the device must first learn your network's normal traffic patterns, commonly called "behavioral baselining." This process takes time, in some cases as long as several weeks.

StealthWatch uses a distributed architecture for deployment, with a master console that communicates with distributed sensors via specified ports and encrypted channels. The management console is not strictly needed for network safety. We installed the M250 alone on a satellite network, and it was quite effective.

StealthWatch doesn't require a separately managed database, and it secures each install by presenting new default command line and administrative login combinations. Another plus is its capability of integrating with Snort 2.0.5 and ISS RealSecure Network Sensor 7.0, allowing you to pull information from these signature-based detection systems into the StealthWatch console.

StealthWatch will appeal to the detail-oriented user. But the level of detail bleeds into the configuration process, which is intricate and time-consuming. On the upside, the appliance has an auto-tuning feature that sets the initial "concern index" threshold high enough to avoid false positives yet low enough to continue monitoring suspicious network activity. Lancope helps organize this monitoring by grouping similar hosts into zones.

Just as StealthWatch needs time to learn your network, administrators will need time to learn StealthWatch. The dashboard is split into two components: A concern index focuses on the sources of attacks, while a target index focuses on the destinations. The StealthWatch appliance monitors the behavior of each host on the network, as well as cumulative network activity. The higher the index value, the more likely a source is dangerous or a target is under attack.

Learning to judge index values and set appropriate thresholds doesn't come easy. Although the dashboard provides a nicely consolidated view of potentially anomalous events, a certain amount of networking expertise is required to interpret what's presented. Ultimately, StealthWatch requires a technically savvy operator and shouldn't be used by a novice administrator.

Lancope does provide a number of features designed to make StealthWatch easier to use. The watch list, for example, allowed us to enter a specific IP or port number to monitor on an ongoing basis. We used the watch list to track the Naval Postgraduate School's e-mail server during the Sasser outbreak. Unfortunately, it's not as easy to specify hosts or ports to ignore; Stealth-

Watch can ignore alarms from specific machines, but that's not quite the same. An easy way to whitelist trusted machines would be a good addition.

Reporting was more than adequate but could be improved. Although we could drill down on alerts to discover details of suspicious activity, we would also have liked to see hyperlinks to graphs in the daily reports, for example, so we could drill down to graphical views of traffic and anomalies.

Overall, we found StealthWatch to be an excellent solution, with one downside — the lack of a signature-based detection engine. Its capability of flagging unknown attacks is a huge benefit, but it requires expertise and interpretive skill from administrators. Although the quick-and-dirty identification of known attacks is valuable, this is made unnecessarily difficult by StealthWatch. Whether by integration or parallel deployment, combining StealthWatch and a signature-based IDS would enhance overall security.

Nevertheless, if we were charged with bringing maximum security to a mission-critical network and money were no object, StealthWatch would be our first choice. Its capability of detecting zero-day attacks and all anomalous occurrences, such as our Sasser worm, move it ahead of the pack.

Border Guard is our No. 2. Combining easy setup, smooth management, and powerful reporting, it brings much-needed polish and an additional measure of effectiveness to a solid Snort core. Border Guard is also an excellent value, making it a close second to StealthWatch for any network.

Joseph L. Roth (joe@javajoe.net) is network security group department head at the Naval Postgraduate School. Mark A. Givens and Charles D. Herring of the Naval Postgraduate School contributed to this review.

The Early Bird Gets the Worm

INTRUSION DETECTION AND PREVENTION SYSTEMS ARE TYPICALLY GENERALISTS, scanning network traffic and alerting you to any kind of threat or anomaly. Arbor Networks' Peakflow X is a specialist, using anomaly-based detection techniques specifically to thwart unknown or "zero-day" worms. If you're running Check Point Software Technologies or Cisco network gear, you can even automate port blocking to choke off propagating worms, while allowing legitimate traffic to pass through.

The Peakflow X solution consists of two hardware appliances: Collector, which monitors traffic, and Controller, which gathers information from one or more Collectors. Collector is not an inline device designed to block harmful traffic, nor is it typically deployed at the perimeter. Arbor suggests deploying Collector at the network core or near the datacenter, where it can monitor communications among many hosts.

Peakflow X's Quarantine Preview shows the breakdown of safe and suspicious traffic flows between hosts. .

Peakflow X focuses on the relationships of machines in the network. It learns which machines talk to which, which ports they use, and so on, ultimately producing a spatial model of normal communications that it uses to flag worm-propagating behavior — the steps a worm takes to seek out and infect other machines across the network.

Peakflow X provides invaluable information for combating a worm attack. The first thing it did when attached to our network was passively map the network. Located on the map is a search button that allowed us to find machines that were communicating using any specific port. The map also displays ports in use and active conversations between hosts.

Peakflow X has a Safe Quarantine function that works with Check Point firewalls, Cisco routers, and Cisco Catalyst 6000 series switches. At the click of a button, Safe Quarantine creates an ACL (access control list) that blocks unauthorized traffic to an identified port while allowing authorized traffic to get through. By mapping port usage and whitelisting authorized traffic, Peakflow X effectively chokes off the worm. Of course, if your network isn't built on Cisco, you'll need to perform port blocking manually.

Arbor's technology is unique, and it gives users a peek at the cutting-edge whitelist prevention systems to come. The $100,000 sticker price won't appeal to budget-conscious shops, but Peakflow X is a darn good worm-defense system, and we look forward to watching this technology mature — and hopefully integrate with a broader range of network gear — in the future. — *V.R.G. and J.L.R.*

Mark A. Givens and Charles D. Herring of the Naval Postgraduate School contributed to this review.

Peakflow X 3.0

Arbor Networks arbornetworks.com

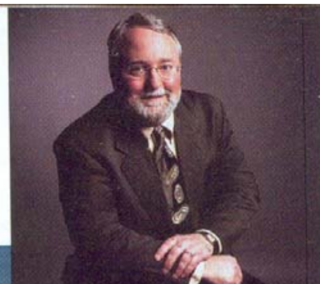
EXCELLENT 8.6

Threat Detection (30%)	9
Management (20%)	9
Ease of use (10%)	9
Scalability (10%)	9
Security (10%)	10
Setup (10%)	7
Value (10%)	6

COST: $100,000 for a typical deployment (Controller and Collector)

PLATFORMS: Management interface: Web browser with Adobe SVG plug-in

BOTTOM LINE: Peakflow X focuses on detecting worm outbreaks. It excels at threat detection, sports a user-friendly interface, and is easy to manage as a distributed system. This solution is expensive to deploy, however, and requires a skilled administrator.



EDITOR'S LETTER

The Luck of the Virus

THERE'S A REASON YOU NEED INTRUSION DETECTION SYSTEMS

WHEN IS A VIRUS ATTACK LUCKY? When it strikes right in the middle of a test of intrusion detection systems. In fact, *InfoWorld* was lucky many times over as we conducted the testing for "Network Detectives: Inspecting the Inspectors" (see page 38). Not only were we slammed by the Sasser worm, but we ran smack dab into a host of Microsoft IIS attacks and a plague of Gator spyware. Needless to say, our review team was pleased; there's nothing quite like real-life attacks on real-world networks to find out what really works.

Luck also played a role in our choice of test venue. Contributing Editor Victor R. Garza was attending a Wi-Fi Planet conference last year when he met Lieutenant Commander Joseph L. Roth, department head of the Network Security Group at the Naval Postgraduate School (NPS) in Monterey, Calif. The two struck up a conversation, and pretty soon they decided to collaborate on a test of vulnerability assessment appliances (infoworld.com/117) at NPS. This week's IDS test is a continuation of that serendipitous partnership with NPS, a facility with 3,000 nodes on its network and a host of top-notch IT talent to watch over it.

For our four-month torture test, we invited the major IDS vendors to participate; a few — including McAfee, Sourcefire, and Symantec — declined because the timing wasn't right. (Look for our reviews of their new releases in coming weeks.) The six products that survived our testing, however, were put through the proverbial mill. Over the course of the trial, our team detected more than 4,000 "events" from nearly 1,000 unique attackers.

By and large, these IDS solutions acquitted themselves admirably, although some used a signature-based approach and others employed anomaly detection algorithms to spot the black hats. Ultimately, our review team concluded that, in the safest of all possible worlds, an IDS would use both signatures and anomaly detection.

Garza and company might have drawn another conclusion, namely that open source folks have more fun. Whereas the commercial IDS products have sober names (Border Guard and StealthWatch), the open-source IDS of choice is Snort (nicknamed "The Pig"). The Pig's most commonly used graphical front end is the colorfully named ACID (Analysis Console for Intrusion Databases). And then there are the porcine-inspired Snort add-ons Barnyard and Oinkmaster.

Snort creator Martin Roesch — founder of security pioneer Sourcefire and an *InfoWorld* 2004 Innovator — confirmed our suspicions about the open source crowd in a post-test conversation. ACID, he confided, is on its way out as the preferred Snort GUI, soon to be replaced by SGUIL. And what does that stand for? Snort Graphical User Interface for Losers, of course.

That's a winner in our book.

Steve Fox (steve_fox@infoworld.com) is editor in chief at InfoWorld.

REACHING THE EDITORS
Editors can be reached via e-mail, fax, telephone, or mail. A list of editors and contact information is at infoworld.com/1157.
E-MAIL: E-mail is routed to individuals' desktops. Please use the following form: firstname_lastname@infoworld.com. Do not include a middle name or middle initials.
TELEPHONE: The switchboard is open weekdays 8:30 a.m. to 5:30 p.m. Pacific time. After 5:30 p.m. you will be directed to individual extensions.
San Francisco office (415) 243-4344 or (800) 227-8365
News, Editorial Projects fax (415) 978-3140
Reviews & Testing fax (415) 978-3275
MAIL: 501 Second Street, San Francisco, CA 94107

C. APPENDIX SUMMARY

This appendix provided a quick synopsis of the complementary IDS comparison and the actual article from InfoWorld Magazine. This author collected no monies from InfoWorld or any of the vendors.

LIST OF REFERENCES

[BACE-00] Bace, R. and Mell, P., "Intrusion Detection Systems," January 2000, http://www.snort.org/docs/nist-ids.pdf, Accessed 4 August 2004.

[BG-03] Border Guard Technical Data-Sheet. Version 4.2. January 2004.

[CSUSB-04] Normal Distribution Calculator http://www.math.csusb.edu/faculty/stanton/m262/normal_distribution/normal_distribution.html, Accessed 4 August 2004.

[DAS-01] Das, K., "Protocol Anomaly Detection for Network-Based Intrusion Detection," August 2001 http://www.sans.org/rr/papers/index.php?id=349, Accessed 4 August 2004.

[DENN-02] Denning, D., "Information Warfare and Security," October 2002.

[DESAI-03] Desai, N., "Intrusion Prevention Systems: the Next Step in the Evolution of IDS," February 2003, http://www.securityfocus.com/infocus/1670, Accessed 4 August 2004.

[DISA-01-01] Defense Information Systems Agency, "Defense in Depth," August 2001, http://iase.DISA-01.mil/ETA, Accessed 4 August 2004.

[DoD-97] Department of Defense Instruction Number 5200.40 "DoD Information Technology Security Certification and Accreditation Process (DITSCAP)." December 1997.

[DoD-03] Department of Defense Instruction Number 8500.2 "Information Assurance (IA) Implementation," February 2003.

[EWEEK-04] eWeek Enterprise Newsweekly Magazine, "Fighting Spam Effectively in the Enterprise," June 2004.

[IATF-02] Information Assurance Technical Framework "Defense in Depth," September 2002. http://www.iatf.net/framework_docs/version-3_1/index.cfm, Accessed 4 August 2004.

[INFOSEC-03] Roiter, N., "Security Switches on Track," November 2003, November 2003 - Security Switches on Track ... Multifunction devices aren't taking the world by storm, but demand for consolidation and performance is fueling interest, Accessed 4 August 2004.

[INFOSEC-04] Plante, A., "Stuffing SPAM," May 2004, http://infosecuritymag.techtarget.com/ss/0,295796,sid6_iss386_art765,00.html, Accessed 4 August 2004.

[MAIRS-02] Mairs, J., "VPN's: A Beginner's Guide," 2002.

[McGRAW-00] McGraw, G. and Morrisett, G., "Attacking Malicious Code: Report to the Infosec Research Council. IEEE Software, 17(5):33 – 41, September/October 2000.

[MOORE-01] Moore, D., et al., "Introduction to the Practice of Statistics," W. H. Freeman and Company, New York, 2001.

[NETSCREEN-03] Netscreen Technologies, "Defense in Depth," March 2003, http://www.netscreen.com/dm/techpubs/downloads/wp_def_in_depth.pdf, Accessed 4 August 2004.

[NIST-03] Mell, P., Hu, V., et al., "An Overview of Issues in Testing IDS's," National Institute of Standards and Technology ITL, July 2003.

[NORTH-02] Northrup, Tony, "Internet Firewalls," August 2002 http://www.microsoft.com/windowsxp/expertzone/columns/northrup/02august12.asp, Accessed 4 August 2004.

[SW-03] Stealthwatch Owners Manual, Version 3.0.0, July 2003.

[YUN-01] Yun, R. and Vozzola, S., "Network Defense in Depth: Evaluating Host-Based Intrusion Detection Systems," June 2001.

[WEB-03] The Webopedia Dictionary Homepage http://www.webopedia.com/, Accessed 4 August 2004.

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California

3. Marine Corps Representative
   Naval Postgraduate School
   Monterey, California

4. Director, Training and Education, MCCDC, Code C46
   Quantico, Virginia

5. Director, Marine Corps Research Center, MCCDC, Code C40RC
   Quantico, Virginia

6. Marine Corps Tactical Systems Support Activity (Attn: Operations Officer)
   Camp Pendleton, California

7. Professor Dan C. Boger
   Naval Postgraduate School
   Monterey, California

8. Professor Alex Bordetsky
   Naval Postgraduate School
   Monterey, California